

# SENIOR DESIGN I

## Sean

Sound Enhancing Autonomous Network



DEPARTMENT OF ELECTRICAL ENGINEERING & COMPUTER SCIENCE

UNIVERSITY OF CENTRAL FLORIDA

Dr. Samuel Richie and Dr. Lei Wei

Final Draft of Senior Design Project

### Group 14:

Annette Barboza

EE

netb@knights.ucf.edu

Ayanna Ivey

CpE

a.ivey@knights.ucf.edu

Brandon Kessler

CpE

brandon.kessler44@knights.ucf.edu

**Advisors/Contributors:** Jonathan Tucker, *Signal and Image Processing*

Wasfy Boushra Mikhael, *Analog and Digital Signal Processing*

## Contents

- 1 Executive Summary**
- 2 Project Description**
  - 2.1 Motivation
  - 2.2 Objectives
  - 2.3 House of Quality
  - 2.4 Requirements Specifications
- 3 Research**
  - 3.1 Market Analysis
    - 3.1.1 Hearing Aids
    - 3.1.2 Wearable Technology
      - 3.1.2.1 Wearable Cameras
  - 3.2 Existing Products
    - 3.2.1 Earbeamer
    - 3.2.2 Here Active Listening
    - 3.2.3 WIDEX Evoke™
  - 3.3 Relevant Technologies and Methodologies
    - 3.3.1 Robotic Operating System
    - 3.3.2 Beamforming
      - 3.3.2.1 Generalized Cross Correlation and Phase Transformation
      - 3.3.2.2 Delay-and-Sum Beamforming
    - 3.3.3 Computer Vision
      - 3.3.3.1 Object Detection - Human Face Detection
        - 3.3.3.1.1 Knowledge-Based
        - 3.3.3.1.2 Hand-Crafted Feature-Based
        - 3.3.3.1.3 Deep Learning-Based
          - 3.3.3.1.3.1 Detection with Deep Learning Neural Networks
            - 3.3.3.1.3.1.1 RCC Based Detection
            - 3.3.3.1.3.1.2 YOLO Based Detection
    - 3.3.4 Mobile Application Development
      - 3.3.4.1 iOS Development
    - 3.3.5 Wireless Communication
      - 3.3.5.1 WiFi
      - 3.3.5.2 Bluetooth®
      - 3.3.5.3 Real Time Streaming Protocol
    - 3.3.6 3D Printing
- 4 Related Standards and Design Constraints**
  - 4.1 Related Standards
    - 4.1.1 IEEE 802.15.1
    - 4.1.2 IEEE 802.11
    - 4.1.3 IEEE 297
    - 4.1.4 IEEE830

- 4.2 Design Constraints
  - 4.2.1 Economic and Time
  - 4.2.2 Manufacturability
  - 4.2.3 Sustainability
  - 4.2.4 Environmental, Social Political, Ethical
  - 4.2.5 Health & Safety
  
- 5 Trade Study**
  - 5.1 Processor
    - 5.1.1 Processor Trade Study
      - 5.1.1.1 Jetson AGX Xavier Developer Kit
      - 5.1.1.2 Jetson TX2 Developer Kit
      - 5.1.1.3 Jetson TX1 Developer Kit
      - 5.1.1.4 Raspberry Pi 3
      - 5.1.1.5 Processor Trade Study Take Away
  - 5.2 Microphone Array
    - 5.2.1 MATRIX Voice
  - 5.3 Cameras
    - 5.3.1 Camera Pixel Evaluation
  - 5.4 3D Printing
  
- 6 Design**
  - 6.1 Hardware Design
    - 6.1.1 e-CAM132\_TX2
    - 6.1.2 Camera Carrier Board
    - 6.1.3 MATRIX Voice
      - 6.1.3.1 Microphone Array
      - 6.1.3.2 FPGA
    - 6.1.4 PCB
    - 6.1.5 NVIDIA Jetson TX2
      - 6.1.5.1 GPU
      - 6.1.5.2 CPU
      - 6.1.5.3 WiFi Module/Bluetooth
    - 6.1.6 User Hardware
      - 6.1.6.1 Phone Hardware
      - 6.1.6.2 Headphones
  - 6.2 Software Design
    - 6.2.1 Audio Subsystem
      - 6.2.1.1 Audio Capture
      - 6.2.1.2 Sound Enhancement Filter
      - 6.2.1.3 User Voice Filter
      - 6.2.1.4 Human Voice Detection
    - 6.2.2 Visual Subsystem
      - 6.2.2.1 Video Capture
      - 6.2.2.2 Human Face Detection
      - 6.2.2.3 Final Video Output

- 6.2.3 Integration of Visual and Audio Systems
  - 6.2.3.1 Align Visual and Audio Space
  - 6.2.3.2 Confidence Analysis
  - 6.2.3.3 Confidence-Influenced Audio Mixing
  - 6.2.3.4 Environment-Influenced Audio Mixing
  - 6.2.3.5 Final Audio Signal Output
- 6.3 Sean Mobile Application
  - 6.3.1 User Interface
  - 6.3.2 Training Mode
  - 6.3.3 User Environment Classification
  - 6.3.4 Other Relevant Features
  - 6.3.5 Speech to Text API
- 6.4 Packaging
  - 6.4.1 Cooling

## **7 Integration and Testing**

- 7.1 ROS and AVIS
- 7.2 Mobile App and Device Communications
  - 7.2.1 Streaming Audio or Video: Wi-Fi or Bluetooth®
- 7.3 Testing
  - 7.3.1 Preliminary Testing
  - 7.3.2 Independent System Testing
  - 7.3.3 Integrated System Testing
  - 7.3.4 Final Product Testing

## **8 Project Operation**

- 8.1 Instruction Manual

## **9 Administrative Content**

- 9.1 Milestones
- 9.2 Budget Analysis
- 9.3 Project Status Diagram

## **10 Results and Conclusion**

### **Appendix A References**

### **Appendix B Permissions**

### **List of Figures**

- 1 Price Points of Hearing Aids from 1991-2008
- 2 CCS Insight Wearable Technology Market 2016 and 2020
- 3 Speech and Noise Combined
- 4 Screenshot of Earbeamer App
- 5 Here Active Listening Earbuds

- 6 Here Active Listening Mobile Application
- 7 WIDEX Evoke™ Mobile Application
- 8 ROS Node and Master Relationship
- 9 Focal Pattern Simulation of Microphone Array
- 10 Localize Source of Sound (1)
- 11 Localize Source of Sound (2)
- 12 Delay-and-Sum Beamforming Visual
- 13 Example of Object Detection Algorithm
- 14 Face Template by Human Created Rules
- 15 Haar Cascades
- 16 Human Neuron Diagram
- 17 Fully Connected Neural Network
- 18 Softmax Function
- 19 Total error influence on the weight updates of a single node
- 20 Model that fits a 2-class problem well using logistic regression
- 21 Convolution Operation Visual
- 22 Detection Example Image with Labeled Data
- 23 Regions with CNN Features
- 24 Image passed Through Fast-RCNN
- 25 Faster RCNN
- 26 Input Image in YOLO algorithm
- 27 YOLO architecture diagram
- 28 Diagram of a Piconet
- 29 Visual representation of a GPU vs a CPU cores
- 30 Image of Jetson AGX Xavier Developer Kit
- 31 Microphone Array Matrix Voice
- 32 Average Human Dimensions
- 33 e-CAM130\_MI1335\_MOD
- 34 ABS Printing Material
- 35 High Level Diagram Of Sean System
- 36 Sean Hardware Block Diagram
- 37 Schematic for Temperature Sensor
- 38 TX2 Carrier Board
- 39 Sean AVIS Algorithm flow
- 40 Sean Final Output
- 41 e-CAM132\_TX2's IFOV
- 42 Sean Camera and Audio Space
- 43 Sean App Introduction Screen
- 44 Sean App Home Screen
- 45 Sean App User Select Screen
- 46 Sean App Training Mode Screen 1
- 47 Sean App Training Mode Screen 2

48	Sean App Training Mode Screen 3
49	Sean Environment Select
50	Sean App Video Mode Screen
51	Sean Packaging Prototype
52	Sean Mounting Hole Diagram
53	Nvidia TX2 Thermal Transfer Plate
54	Nvidia TX2 Active Cooling Fan
55	USB Cooling Fans
56	Sean LED Indications
57	Project Status Diagram

### List of Tables

1	Engineering-Marketing Trade-off
2	Requirements Specification
3	Bluetooth® Version Improvements
4	Requirements vs Processor Features
5	Jetson AGX Xavier Developer Kit Specs
6	Jetson TX2 Developer Kit Specs
7	Jetson TX1 Developer Kit Specs
8	Raspberry Pi 3 Specs
9	Microphone Array Options
10	e-CAM132_TX2 Specs
11	e-CAM20_CUTX2 – 2MP Specs
12	LI-JETSON-KIT-IMX477-X Specs
13	Camera Requirements
14	Audio Confidence Weight Values
15	Visual Confidence Weight Values
16	Confidence Influenced Source Gain Tiers
17	Dimensions of Sean Packaging
18	Dimensions of Components
19	Preliminary Testing
20	Independent Visual System Testing
21	Independent Audio System Testing
22	Independent Mobile Application System Testing
23	Integrated System Testing
24	Final Product Tests
25	Senior Design I
26	Senior Design II Milestones
27	Actual Budget

## **1 Executive Summary**

Sound Enhancing Autonomous Network (Sean) will be a portable device that a user can employ to enhance the sound of another human's voice while eliminating background noise, including other human voices. The system will consist of a small camera and a microphone array to record visual and audio input that will be aligned in a common angle space. The camera will be able to clearly record another person's facial features at a conversational distance and be able to identify other people within each frame at a further distance. The microphone array will be able to gather audio data, and Sean can filter background noise and identify which direction sources of audio are coming from. A product such as this is not only helpful for people who may be hard of hearing, but can also increase the ease and quality of life for people who frequent environments that have background noise. Hearing is one of the five basic human senses used in everyday life, and devices that help individuals hear better have a high market value. Naturally, many companies are more focused on profiting from the basic human need rather than actually trying to benefit those that are suffering with a hearing disability. Consumers cannot benefit from a product if they cannot afford it, so this project will have a market value significantly less than the average for hearing aids currently. This will allow for a wider audience to benefit from it, rather than being targeted towards groups with a large disposable income. Not only will it be cheaper than basic hearing aids, it will also provide more functionality and receive input from the user, which makes the low cost even more favorable. As previously stated, Sean will filter out sound as opposed to basic hearing aids that simply amplify all sound, including unwanted background audio. In order to increase accessibility and ease of use we will interact with the device through the use of a smartphone application which will be created to power the device on and off, control the volume the user will experience and stream video from the camera to the user's device. This will be beneficial for the user since many consumers that would be interested in this device use smartphone applications in their daily life already. The device will be roughly the size of a sheet of paper with the thickness of no more than 3in. The device will be battery operated in order to make it portable and to enable the possibility of it being wearable. The largest components are the processor and battery which both require additional space in order to ensure neither overheats.

## **2 Project Description**

Sean provides a way for consumers to be able to filter out background noise and outside conversation to focus on the sounds that are important to the user. This section will discuss the motivation behind developing a project based on helping those with hearing disabilities and users who are just interested in a directed sound enhancing experience. We explore why someone would prefer this device over the ones currently available on the market. The objectives and requirements specifications for this project will be discussed as well.

### **2.1 Motivation**

The direction technology has been taking recently makes the lives of consumers more convenient in *almost* every conceivable way. [1] However, there have been very few attempts to make products for those who live with disabilities as accessible as products that are simply for convenience. In the case of hearing impairment, approximately 15% of American adults report having some trouble hearing.

Hearing loss presents itself in three different types: conductive hearing loss, sensorineural hearing loss, and a mix of both. In permanent cases, conductive hearing loss generally affects the overall loudness of a sound, while sensorineural can affect loudness and perception of tone [4]. For the majority of the people who live with any of these, their options are limited to potentially invasive procedures in an attempt to correct the hearing or using hearing aids. Hearing-aids are one of the most widely available removable solutions to hearing loss. [2] They help users by converting sound to digital signals, amplifying those signals, and passing the amplified signal back to the user as sound. One of the problems the hearing aid has a hard time with is effectively separating wanted and unwanted noise to only amplifying what is wanted.

Our solution to the aforementioned issues is to design and create a noninvasive alternative to hearing aids, the Sound Enhancing Autonomous Network or Sean. Sean utilizes deep learning approaches to detect people in view of the user and combines these decisions with digital signal processing on the audio signals to focus on intended human voices. The goal is to build a portable device that users can easily use to accurately amplify the sound they want to hear in Bluetooth® connected headphones or earbuds. Sean raises the intensity of voices and diminishes background noise which helps our targeted users: people who have permanent conductive hearing loss and/or sensorineural hearing loss. However, Sean can be of use to anyone interested in a directed sound experience.

Sean aims to improve the quality of life of those who are hearing impaired by replicating the experience of being able to focus in on a person speaking in an indoor environment. This device works in real-time to lower the background noise in an indoor environment with a moderate to high signal-to-noise ratio to clearly make out what a person is saying. Sean will use computer vision to help detect when a person is in view of the user and digital signal processing methods to separate the background noise and the voice of the detected person. This will consequently lower the amplitude of the background noise and raise the amplitude of the voice so that it becomes the dominant signal.

## 2.2 Objectives

The main goal for this project is to create a smart alternative to the common hearing aid. The system will make decisions on when to amplify and attenuate different sources of sound based on information provided from both the visual and audio domains. Effectiveness and efficiency are two main components that guided the decisions in framing the scope of the system. Sean must enhance a user's experience when it comes to the quality of the sound he or she receives. Keeping these main tenets in mind, the following core objectives relating to the function of Sean have been established:

- Predict the human voice source of sound the user wants to hear out of a range of potential sources and amplify that sound
- Predict sources of sound that the user does not want to hear, including background human voices and attenuate those sounds
- Provide a mobile application interface in which the user can interact with Sean
- Give a visual output of how Sean is making decisions via a live stream to the mobile application



- Create a comfortable wearable platform for Sean that can illustrate the potential of producing a portable system.
- Improve the quality of life of hearing impaired people as well as providing an enhanced experience for average users
- Be a user-friendly and intuitive product

### 2.3 House of Quality

Shown in Table 1 below, the engineering and market requirements are compared to identify potential trade-offs that might need to be made in the design of Sean. This is necessary in order to ensure our design will be realistic from both a development and a consumer perspective. It will also aid the group in prioritizing the project goals to ensure all of the essential needs are met.

1. Engineering Requirements - ■
2. Marketing Requirements - ■
3. ↑↑ - Strong Positive Correlation
4. ↑ - Positive Correlation
5. ↓ - Negative Correlation
6. ↓↓ - Strong Negative Correlation
7. + - Positive Polarity
8. - - Negative Polarity

Table 1 -- Engineering-Marketing Trade-off Table									
		Efficiency	Output Power	Implementation Time	Weight	Cost	Dimensions	SNR	THD
		+	+	-	-	-	-	+	-
High Power	+	↑↑	↑	↓↓	↓↓	↓	↓↓	↓↓	↓
SNR	+	↑	↓↓	↓↓	↑	↑↑			↑↑
Cost	-	↓↓	↓	↓↓	↓↓		↓	↑	↑
Latency	-	↑	↓	↓↓		↓↓		↑	↑
Portability	+	↓	↓	↓↓	↑↑	↓	↑↑	↑	
User Friendly	+	↑	↑	↓	↑↑		↑↑	↑↑	↑↑
Accuracy	+	↓		↓↓		↓	↑	↑↑	↑↑
Resolution	+	↓	↑↑	↓		↓		↑↑	
Engineering Requirement Targets		≥ 70%	≤ 30W	≤ 8 weeks	≤ 5 lbs	≤ \$1800	≤ 93.5x 67.4x 42.5 (mm)	< 60 dB	< 1% @ 95 dB SPL

## 2.4 Requirements Specifications

This section will outline the components of Sean and the specific requirements associated with each part or feature. Specifications are determined by the necessities of algorithms and to ensure a smooth integration of parts.

*Any specification preceded with "Option:" are being considered and will be decided based on feasibility and time constraints .*

Table 2 -- Requirements Specification		
Component	Feature	Specification
Cameras	Forward facing camera with respect to the system	30 frames per second (fps) minimum capture rate
		720p minimum resolution
		60 deg minimum diagonal Field of View (FOV)
		44.3 deg minimum horizontal FOV
		25.8 minimum vertical FOV
		Compatible with chosen processor
Cameras	<i>Option:</i> Backward facing camera with respect to the system	30 frames per second (fps) minimum capture rate
		720p minimum resolution
		60 deg minimum diagonal Field of View (FOV)
		44.3 deg minimum horizontal FOV
		25.8 minimum vertical FOV
		Compatible with chosen processor
Computer Vision (CV)	Autonomously detect human	10 fps minimum processing

Algorithms for Human Detection	within the FOV of the camera	rate
		Detects humans up to 20 feet away
		Correct detection rate of 90%
	<i>Option:</i> Autonomous lip reading recognition	10 fps minimum processing rate
		Reads lips of one human up to 5 feet away
		Correct reading rate of 80%
	<i>Option:</i> Voice generation	10 fps minimum processing rate
		Generates voice of one human up tp 5 feet away
	Processor	Embedded for real-time processing (Including Development Kit)
1.1 lb maximum weight		
7.1 in. x 7.1 in maximum size		
4-core @ 1.5 GHz minimum CPU		
180-core minimum GPU		
4GB minimum memory		
Bluetooth® 4.0 or greater enabled		
Latency of no more than 30 ms		
Microphones	Array of microphones to convert sound to digital	8-20 MEMS microphones in array

	signals	<p>~ 26 dB @ 94 dB SPL (normal for digital microphone)</p> <p>Omnidirectional</p> <p>~60 dB SNR</p> <p>Operating frequency range: 125 Hz-8kHz (average for CIC hearing aid)</p>
Digital Signal Processing Algorithms for Signal vs. Background Noise	Standby State- no cue from CV algorithm	Stay idle allowing only noise cancellation from digital signal processor when no humans are present allowing background noise to sound natural and non-intrusive
	A third state?	
	Operating state-- cue from CV algorithms	Use beamforming to locate source of sound and amplify it while simultaneously lowering the background noise
Digital Signal Processor	Beamforming (up to 20 dB attenuation)	
	Audio sampling rates 8 kHz to 216 kHz	
	Linear phase FIR filter	
	Noise suppression (up to 20 dB attenuation)	
Power Supply	4 hours of continuous power to entire system	
	Compatible with all hardware components	

iPhone Application	Connected through Bluetooth® 4.0 or greater	
	Controls Volume	
	Controls Sensitivity	
	<i>Option:</i> ability to choose individuals to listen to	
System Housing	5 pounds or less	
	Entire system contained	
	Wearable system	Attached to chest

### 3 Research

In this section, the existing solutions, methods, and algorithms used to better hearing aid technology will be discussed. The primary technologies to be discussed, Earbeamer, Hear Active Listening, and the WIDEX Evoke™, will be reviewed and analyzed as they were created with the intent to solve the same major problem that Sean will, the cocktail party problem. The cocktail party problem refers to the difficulty of focusing on a source of sound while trying to effectively filter out the rest of the noises. Earbeamer, Here Active Listening and the WIDEX Evoke™ were chosen as they use similar methods and practices as Sean will and will have results and/or user feedback available to take into account when researching and designing. The methods and algorithms researched to solve this will also be included in this section. This includes using computer vision and acoustic beamforming together to produce better, more accurate sound. Ease of use for consumers must be considered as well, so the use of a smartphone application will be explored. In order to have a product that can be used in day to day activities, the product must also be packaged in an appropriate way, which includes the possibility of being a wearable device. In order to ensure smooth communication between the application and Sean research must be done on the use of applications for listening device to ensure this will meet the needs of the user. Therefore mobile application development, and more specifically iOS development, will be discussed in this section. Then methods of communication must be explored to determine which is appropriate for the needs of the project. Both Bluetooth and WiFi will be explored as methods of transmission.

#### 3.1 Market Analysis

Sean is being designed as a noninvasive alternative to hearing aids that will appease problems many hearing aid users experience, for instance: shortened lifespan, stunted amplification, incompetent automatic noise level adjustment, and inflated prices of hearing aids. These will be improved by Sean by merging visual and audio spaces to make the system more accurate and designing it to be wearable device to improve lifespan. With this in mind, it is only appropriate

to conduct a more in depth market analysis of the hearing aid and of wearable technology to design Sean appropriately.

### **3.1.1 Hearing Aids**

[26] According to a report by Zion Market Research, in 2017 the global hearing aid market was estimated to be worth 6.32 billion USD and is estimated to be 9.17 billion USD at the end of 2024. The demand for hearing aids, especially improved ones, has been growing rapidly as hearing loss problems in the population have begun to increase. With several types of hearing aids available, what is crucial now is that the technology of these devices drastically improves to fully meet the needs of current and future users.

There are 5 major types of hearing aids: mini-behind-the-ear (mBTE), behind-the-ear (BTE), completely-in-canal (CIC), in-the-canal (ITC), and in-the-ear (ITE). All of these with the exception of the ITE are privy to wax and moisture build up potentially shortening the life of the device. Many users prefer low visibility of the hearing aid when it is worn; about 71% of surveyed hearing aid users use mBTE hearing aids as they are comfortable and barely visible. Since this means the receiver is inside the ear canal, wax and moisture affect the lifespan of the device and it stunts amplification. This is a common issue with other hearing aids as well that have any parts inside the canal. [27]

In addition to wearability, the way in which it processes and responds to noise in an environment is a crucial factor when someone is purchasing a hearing aid. [27] In a survey conducted by Consumer Reports, 42% of hearing aid users identified one of the most important features to be automatic noise level adjustment. Unfortunately, this feature in most hearing aids is not suitable in every environment and needs to be manually adjusted in whatever program settings are available. This could potentially be improved by adding deep learning capabilities to hearing aids. It would close what is now a very large gap of environments current hearing aids do not function well in and would become better suited to work well in a variety of environments rather than just low noise and low echo environments. It is not to say solutions like this have not already been designed and introduced to the market, but, since other companies have not yet followed suit, this technology is especially expensive and not easily accessible for hearing impaired people to purchase.

[29] Even standard digital signal processing hearing aids are expensive and a large investment for the average American; the average price of a single hearing aid is \$2,300 and it should be noted that customers generally need two. As seen in Figure 1 below, the prices of hearing aids keep increasing despite the fact that the introduction of digital signal processors introduction in other industries has led to large decreases in prices of those products.

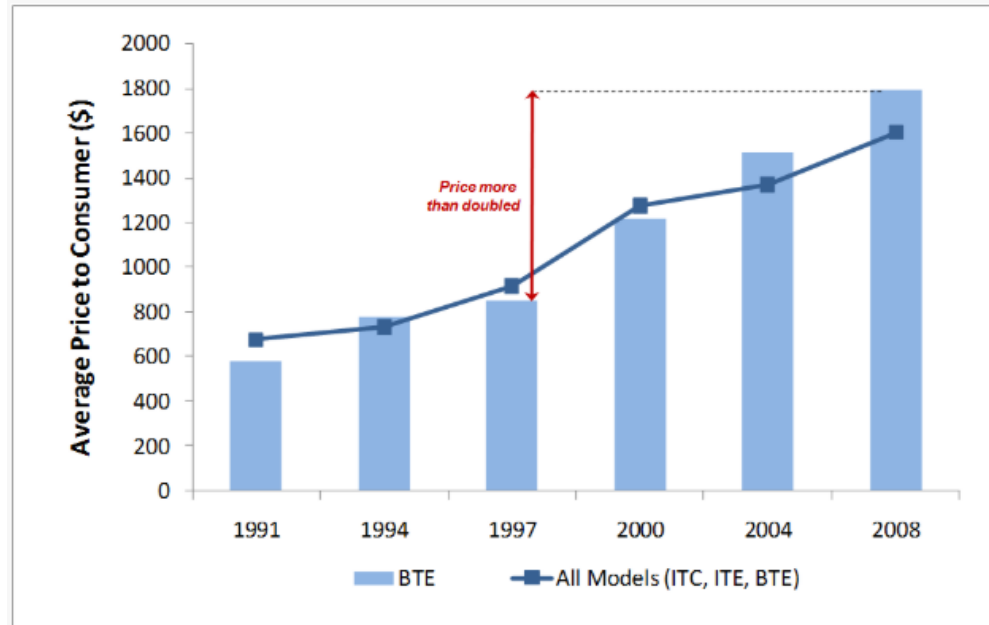


Figure 1. [30] Price points of hearing aids from 1991-2008.

[29] This is likely due in part to six global manufacturers controlling 90% of the market, leaving consumers limited on their options. This allows for them to push the products they believe will sell more and raise the prices unnecessarily. [31] Consumer Reports, with the verified data available, found that the wholesale price of hearing aids had an average retail markup of 117%. With the introduction of intelligent signal processing for hearing aids and this massive predicted growth in the hearing aid market, the demand will allow for smaller or new companies to offer more competitive prices and normalize this new technology, hopefully lowering prices across the board to make great technology accessible to everyone who needs it.

### 3.1.2 Wearable Technology

In a world filled with emerging technology, having interesting features for devices is no longer enough. Consumers are not just concerned about what a device does but also how it looks. With the level of innovation taking place today, it allows processors and chips to be small enough to fit in the palm of a person's hand. This has caused everything from glasses, which people actually need in order to see, to everyday items like a watch to be an accessory. Because of this it has recently become a popular trend in which technology must be fashionable or it will not sell. The simplest illustration of this is with phone colors and cases. The excitement about mobile phones used to just be the fact that it existed and the features it contained. Now when consumers buy phones they are concerned about the color, the size, the eye-catching cases they can put on it, as well as many other things. This trend has shown itself even in timeless basic technologies, such as headphones or watches, as well. It has become such a popular thing to do that now name brand clothing companies are partnering with technology companies to create their own high end, name brand wearable technologies. In fact, the market for wearable technology is expected to be worth \$34 billion by the year 2020. Specifically for hearing devices and wearable cameras, CCS Insight has projected there will be 9 million and 25 million devices sold, respectively. Not only is it necessary for the wearable technology to be fashionable, it must also communicate easily with other devices. This can be done through a direct connection or through the use of

mobile devices. The Apple Watch® is a prime example of wearable technology that also includes the use of mobile applications. The applications downloaded on an iPhone® can be synced and used on the watch. It also syncs the information so that all the music, messages and fitness information that is on your phone can be seen on your watch at the same time. This adds yet another requirement for mobile applications which would be the ability to use it appropriately across multiple devices.

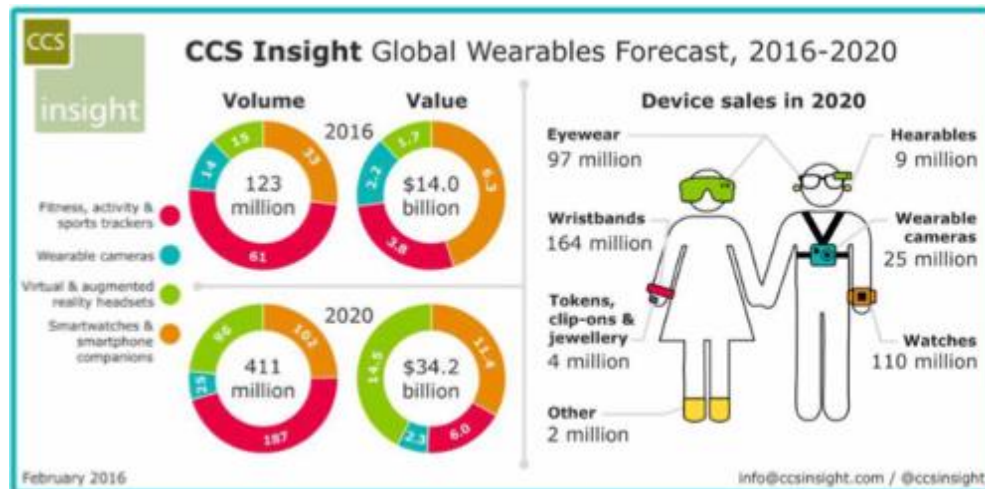


Figure 2. Image above displays a wearable technology market comparison between 2016 and a projection for 2020

### 3.1.2.1 Wearable Cameras

The main driving force for the wearable camera market is the use of social media. With the increase of focus on pictures and video, as well as features such as live video streams, the demand for wearable cameras extremely high and is still increasing. The main categories for wearable cameras are head mounted, body mounted, ear mounted and camera glasses. The issue with wearable cameras is stability. If someone cannot accurately process the video that comes out of a camera then it completely defeats the purpose. Stability and automatic adjustments according to environment are imperative for cameras that are constantly moving on a user.

## 3.2 Existing Products

Hearing disabilities are not a new problem for people at all, in fact it has a large market. Outside of marketing toward people with disabilities, most people listen to music so the push for a high sound quality in headphones and speakers has always been present. Therefore there are many existing products that target consumers with hearing disabilities or wish to reduce noise in their environment. As previously mentioned, there have been solutions and hearing aids developed using intelligent signal processing techniques to improve upon the already existing hearing aid design: beamforming, speech recognition, computer vision, etc. Since Sean will be have some of these techniques implemented in its own design it is only fitting to analyze products and solutions that use these. This section will analyze the Earbeamer, Here Active Listening, and WIDEX Evoke™ and how they have improved upon the standard hearing aid or aided in the push towards a world with the best possible quality of sound.



### 3.2.1 Earbeamer

Although some higher end hearing-aids can take the user's environment into account and try to reduce noise, a common complaint among hearing-aid users is that all sound, including unwanted background noise, is amplified and does nothing to help them hear what they want to hear as displayed in Figure 3. A few groups have attempted to correct this same problem. Gupta et al. from the University of Massachusetts approached this problem in their senior design project. Their solution allowed the user to independently control the volume of different people in a room. They used an Xbox Kinect to detect individuals and a microphone array to perform beamforming and assign sound to a person while eliminating noise. However, their system was stationary and could only work in the room it was calibrated to. [28]

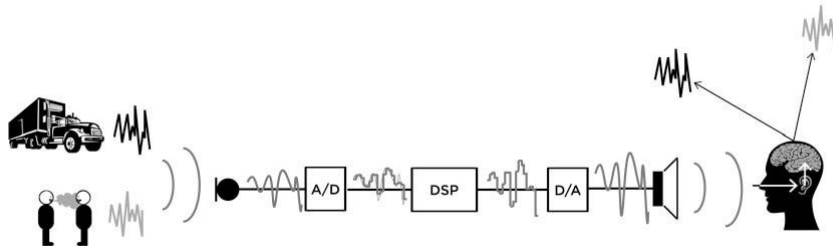


Figure 3. [8] Example of speech and noise being combined and confused during processing. This leads to the amplification of both signals and hard for the user to hear the sought after conversation.

The two major components in this project were the Xbox Kinect and the microphone array. The Xbox Kinect served as the visual aid that was able to detect sources of sound--people talking--in a room where the entire system is set up and determine the location of them relative to the microphone array. The microphone array was built and designed by the group; they decided that a nested linear array of 16 microphones was most ideal for their design. The group weighted their microphones so that the array response main lobe was as narrow as possible and the sidelobes were as small as possible to properly implement their delay-sum beamforming algorithm and then included RC filter to eliminate aliasing--potential distortion from out of range frequencies being mapped to known frequencies in an output. [28]

It was decided by the group to use an iPhone application as the user interface to pick a target for amplification or attenuation. This shows users where the targets are relative to the location of the microphone array and Xbox Kinect--which seem to be represented by the black rectangle on the left of Figure 4 shown below. Two-way communication was implemented using WebSockets; the processing computer connects to the application via WebSockets and sends 2D coordinates from the Xbox Kinect of where the target is about once per second and maps it accordingly on the application while simultaneously taking any user input (a tap on a target on the mobile app) and responding accordingly to it. It was found that through this system the latency of switching from target to target was negligible. [28]

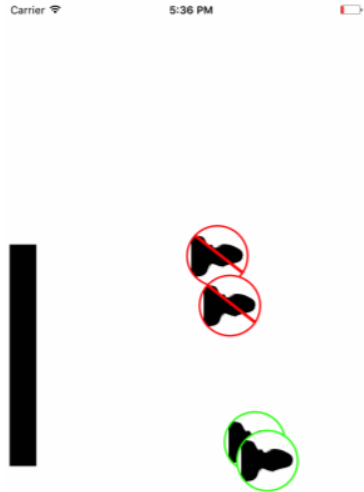


Figure 4. [28] iPhone screen sample of Earbeamer application showing which can be assumed to be showing which targets have been amplified (green) and which have been attenuated (red).

[28] The group met one of their primary objectives: selectively amplifying nearby targets, giving the user the ability to turn the volume up or down on people in the room. However, it is not mentioned in their results whether or not this was a success for any hearing impaired person, which would be a valuable data for future researchers. [28] In their conclusions they noted that the main drawbacks of their system included latency, beamwidth, and audio quality; they mention that an FPGA with the FIR filters and beamforming algorithms could have alleviated the processing overhead, and analog-to-digital converter with more channels could have allowed them to use more microphones to capture more of the human speech frequency spectrum without trading off beamwidth, and an algorithm dedicated to noise cancellation could have improved the audio quality.

The implementation of this design shows others researching this that adding this visual space has helped the user immensely by locating all potential sources quickly and allowing them to choose what source (voice) they want to hear. Merging the two spaces could have potentially increased the accuracy in their project, but it does not seem to be needed as this was designed to stay in a stationary space and works well in that regard despite the audio quality not being what they had hoped.

### 3.2.2 Here Active Listening

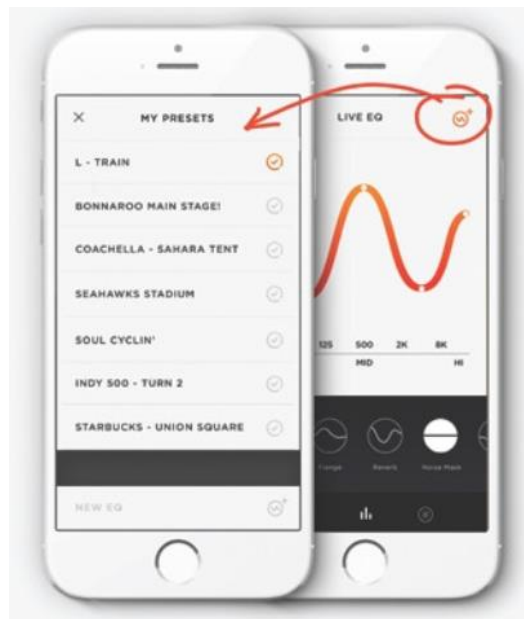
This device is similar to SEAN in that it takes in all the noise in an environment only plays back what the user wants to focus on. It consists of two wireless earbuds and a smartphone application which interact through a Bluetooth low energy 4.0 connection. The microphone used to collect audio is a MEMS omnidirectional microphone. The battery life of the earbuds is 6 hours and it comes with a compact charging case that will hold two full charges. The device is compatible with any iOS and any device that supports Android 4.0 and up. Instead of mainly catering to users who may have difficulty hearing this device enhances the day to day experiences of people who may have no trouble hearing at all. This mimics what we would like the do with Sean's app in that it allows you to make real time adjustments to what the user is hearing, such as sound

level. For this device, it allows the user to do things such as tune out a crying baby or even adjust bass levels to music that may be played around someone. It is currently listed at \$200. When compared to the prices of today's wireless headphones such as Apple AirPods or Beats by Dr. Dre which are listed at about \$150 and \$300 respectively, this seems to be a reasonable price for a listening device that could be used almost everyday.

After the initial launch of the device Doppler Labs added a "stretch goal" of being able to save sound settings within the device and access them to be used later. Doing will create a truly customizable user experience. It will also minimize set up times for a user because they will not have to take to the time to readjust their settings to an optimum level every time they decide to use the system. The integration and use of a mobile application to store custom voice profiles would also be a stretch goal for Sean.



*Figure 5. Displays Doppler Labs' Here Active Listening earbuds that allow the user to adjust sound settings for their environment in real time*



*Figure 6. Displays Doppler Labs' Here Active Listening mobile application that controls the earbuds shown in Figure 5. This displays the ability of the application to store sound settings for a user and allow them to be accessed and used later.*

This device uses a number of signal processing algorithms within certain frequency ranges in order to produce the desired sound output. Sound is expected to receive with a delay of less than 30 microseconds which will not be detectable by the user. For testing they used one of the largest music festivals, Coachella. They did a number of tests in various environments such as a small tent filled with people and a lawn filled with thousands of people to ensure the device could function no matter the environment the user was in.

### 3.2.3 WIDEX Evoke™

The WIDEX Evoke™ is a hearing aid platform that uses a machine learning as a solution to hearing aid parameter personalization. This machine learning approach is called SoundSense Learn and was introduced exclusively on Evoke™ products May 2018. [31]

[31] SoundSense Learn allows users to have an increased sound quality and improved hearing aid experience by using their auditory intention--which refers to what they wish to gain from an auditory-related task. This makes assumptions on what is important for the user to hear and amplifies them automatically. WIDEX has made an app that connects to the hearing aids and allows for user control as well as user set up. [31] The calibration of the hearing aids has the user compare two parameter settings and pick which they would prefer in their day to day for all 20 situations WIDEX has provided. Sample screens of the calibration are shown below in Figure 7. These settings are to be decided by the user to make it as personalized as possible for the system to be able to come up with the correct assumptions when it comes to the user's auditory intention.



Figure 7. [31] Mobile app screenshot samples of calibration for WIDEX Evoke™ hearing aids.

This implementation--given its accuracy-- would eliminate a lot of frustration and the number of visits to an audiologist to tune the hearing aid accordingly since the system would do that automatically. Since this is completely reliant on an audio space, it is limited on solely what patterns of audio intention the user previously denoted, so would consequently fail to recognize what a user wants to hear with too many variations of audio intention in for one environment. This could potentially be improved with an analysis of the visual space as well. Given that it starts to fail, much like a standard hearing aid, the user is required to go visit an audiologist to correct any issues or failures in the system that they cannot correct themselves.

[31] Townend et al. from Widex A/S put the SoundSense Learn system through thorough tests due determine what level of improvement it would have in sound quality and comfort in the hearing aid parameter settings. [31] They used all of their own products and systems to test three types of parameter settings against each other: no environment detection system, environment detection system, and SoundSense Learn. [31] Of the 19 test participants that completed the test, 84% preferred the hearing aid parameter settings they achieved with the SounSense Learn for comfort and 89% preferred the hearing aid parameter settings they achieved with the SoundSense Learn for sound quality.

With the overwhelmingly positive results achieved from this there are plans to introduce virtual reality into this to improve the audio intention and make it more precise.

### **3.3 Relevant Technologies and Methodologies**

Since there are not currently any existing products that will tackle the issue of amplification of unwanted noise and the cocktail party problem the way Sean will, all related research will be detailed in this section. ROS, beamforming, machine learning, computer vision, and mobile application development will be discussed in detail as the algorithms to implement them will be used in Sean.

#### **3.3.1 Robotic Operating System (ROS)**

ROS is not so much an operating system as it is more a framework and set of tools available for hardware abstraction, device drivers, communication between processes over multiple machines, and tools for visualization and testing. Since Sean will be performing algorithms in parallel on two different devices--that will eventually communicate with each other after major environment analyzation is complete--ROS becomes imperative to research and to implement in our solution to alleviate any complexity that might exist without it. [57]

ROS is a BSD-licensed system used to control robotic components from a PC. The way the system works is by having an ROS *Master* device that acts as the single point of contact for all ROS *Nodes*--pieces of software--so that they may have the link to find communicate with each other directly. All nodes must be linked with or registered to the Master, which holds all the information about where to send messages to other nodes. With these nodes registered or connected to the master, they can publish information that other nodes can either subscribe--wait for and use information published--or implement services--which allows a node to request data from another node. This is shown in Figure 8. [58]

A network of communication can be created by defining these publisher/subscriber connections and thus assemble a complex system of well-known solutions to small problems that can communicate with each other to solve a bigger problem. This implementation allows users of ROS to multiplex outputs of multiple components into one input for another component, allowing for algorithms to be executed in parallel and allows for the use of different programming languages among the devices used by using the proper connections between nodes. [58]

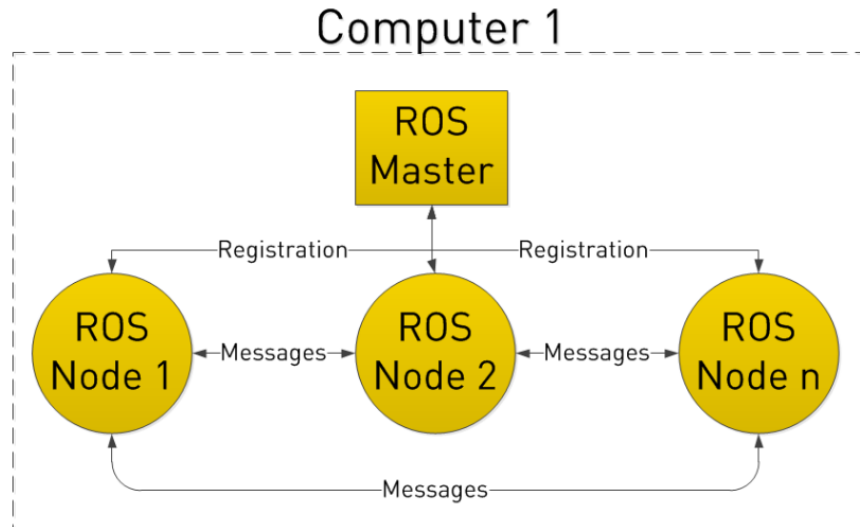


Figure 8. [58] Node and Master relationship for ROS.

### 3.3.2 Beamforming

[34] Acoustic beamforming is a method of sound source localization of a specific sound source from a mixture of other insignificant sound sources based on direction of arrival. It is starting to become common in the hearing aid industry as it improves SNR and is becoming increasingly common in voice detection applications.

Microphones can be configured into different depending on what kind of beam pattern is needed which would be telling of what kind of application it is needed for. Using several omnidirectional microphones will amplify the sound from a source and attenuate all other unwanted noise by focusing the beam at the source. The distance between each of the microphones has a direct relation to spatial aliasing--distortion in the signal due to an insufficient sampling rate. This kind of effect redirects frequencies that are "out of bounds" to know frequencies creating distortions and inconsistencies in the output signal. This can also be corrected with an anti-aliasing filter but generally designers try to account for this in array design and geometry.

In Figure 8 below, the theoretical performance of a uniform circular array composed of 20 microphones is shown on a 3D plot. Dale Grover, of Michigan State University, wrote a program plotting theoretical microphone array sensitivity patterns. The first four meters of data is omitted as it is assumed there are no sources of noise in that range and all of the measurement in the plot are in meters. It can be seen that the cone-like beam converges at about 6 meters in the y direction which allows the assumption that the source of sound or target of the beam in this simulation exists somewhere in the maxima. [34]

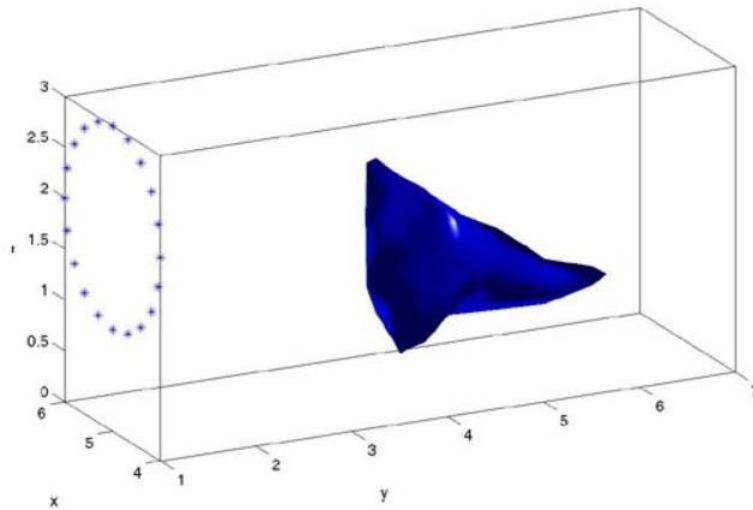


Figure 9. [3] Focal pattern simulation of uniform circular 20 microphone array.

[35] [36] The preferred array configuration is a planar array since it will cover a range of 360° azimuth and a uniform circular array configuration will be used as it will reduce noise from the floor and ceiling while amplifying sounds at mouth level. This can be seen in Figure 8 above.

The beamforming process will need the source position before it can filter the audio to receive the desired sounds. The source can be localized by using the Generalized Cross Correlation-Phase Transformation (GCC-PHAT) algorithm and Delay-and-Sum Beamforming, one of the most basic and simple beamforming algorithms, will be used to shift the signals at each microphone output so that the desired source's signal in each output is aligned. [35]

### 3.3.2.1 Generalized Cross Correlation and Phase Transformation

[34] Determining the location of the source requires that the direction-of-arrival (DOA) is found using the GCC-PHAT algorithm. [35] The cross correlation operation is generally used to detect similarities between two signals and calculate the time of arrival (TOA) or time at which both signals match the most. The TOA is likely to be different for each microphone as they are located different distances in regards to the source. [34] The time difference of arrival (TDOA) is the difference between the time of arrivals for two microphones placed in different locations. A hyperbolic curve representative of the plot between time lag and time is shown in Figure 10 and shows that all point on the curve act as a source of sound. In Figure 9,  $m_i$  and  $m_j$  are the vector locations of microphones 1 and 2, respectively, and time difference has a function of  $p$ , the location of the source is

$$\tau_{12}(p) = \frac{\|p - m_i\| - \|p - m_j\|}{c}$$

where  $c$  is the speed of sound.

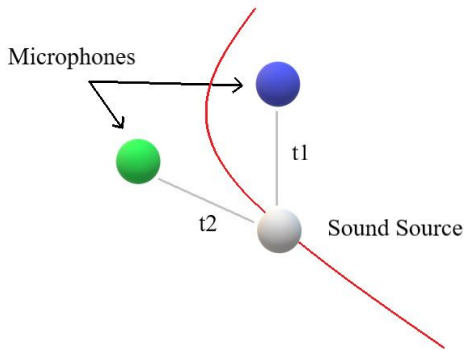
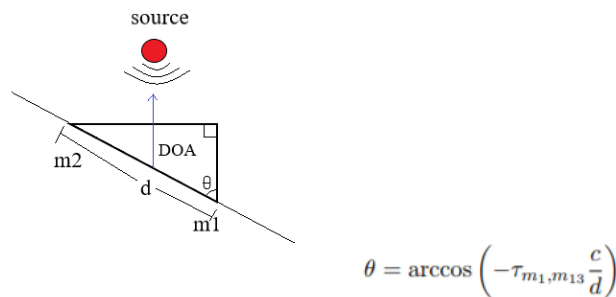


Figure 10. Determining TDOA to localize source of sound.

The argument of the maximum of the cross correlation function below is a unique value of  $\tau$  that represents the TDOA.

$$R_{12}(\tau) = \frac{1}{2\pi} \int_0^{2\pi} \psi_{12}(\omega) X_1(\omega) X_2(\omega) e^{j\omega\tau} d\omega$$

$\Psi(\omega)$  represents the weighting function. Once  $\tau$  is calculated, it is plugged back into the first equation to solve for position p.



$$\theta = \arccos \left( -\tau_{m_1, m_{13}} \frac{c}{d} \right)$$

Figure 11. Determining TOA to localize source of sound. The m2 in the figure is m13 in the equation.

GCC-PHAT can also be utilized to solve for the TOA and use triangulation--trigonometric formulas--to solve for the DOA. [37] This is what the MATLAB function does to solve for GCC and is what will be used to simulate our uniform circular 8 microphone array in Section 5.1.1.

However, there are some issues with generalized cross-correlation especially in real-time applications; cross correlation is cannot account for any deformations in the signal and must be normalized.

### 3.3.2.2 Delay-and-Sum Beamforming

The premise of the delay-and-sum beamforming algorithm with respect to microphone array applications is adding up the like signals of a source from different microphones to produce more spatial samples and thus more accuracy and a better SNR in the output.

To do this, the source direction needs to be known prior, which is covered in the previous section with GCC-PHAT. [35] This angle allows the closest microphone to the source and its azimuth to



be found allowing for all time delays for the other microphones to be calculated. The time delays are due to the distances between each microphone and the direction of the signal from the source.

The time delays are a function of the microphones' angles which are compared against the angle of steering--the direction of the source-- to find the closest microphone angle. Once all the time delays are known, complex weights are applied to each microphone output which account for the delay and shift the signal "aligning" the desired signal in all microphone outputs. This "steering" of the array is shown below in Figure 11. [35]

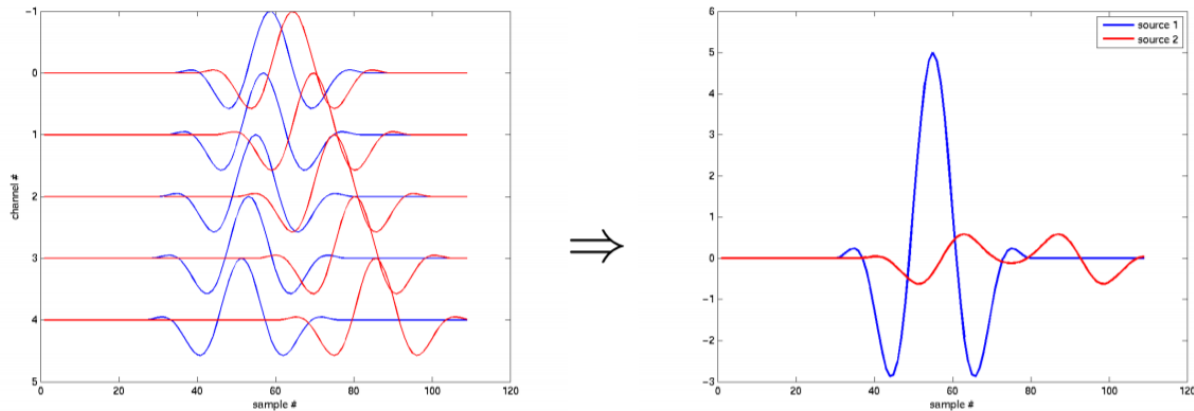


Figure 12. [13] Shows weighted microphone outputs aligned (right) and the final summed signal(left).

When all of the shifted outputs are summed the desired signal is added constructively while noise/interference is added destructively.

$$z(t) = \sum_{m=0}^{M-1} w_m y_m(t - \Delta_m)$$

[36] The definition of the delay-and-sum algorithm is shown above where  $z$  is the beamformer output,  $m$  is microphone number,  $M$  is the number of microphones,  $w_m$  is the amplitude weight,  $y_m$  is the signal and microphone number  $m$ , and  $\Delta_m$  is the delay of the signal at microphone number  $m$ .

### 3.3.3 Computer Vision

[11] Computer vision is the scientific field which involves the automatic extraction and analysis of useful data from an image or video. To be a complete robust system, Sean will take advantage of a visual domain. Information gathered here will be merged with the audio information to make decisions about how many people are in the scene and where they are located.

One of the biggest problems Sean will face is multiple people talking at the same time. With more than one person talking at once, the wrong voice could be amplified or several voices could be amplified and noise would remain present. Sean will solve this problem using a computer vision approach known as object detection.

### 3.3.3.1 Object Detection - Human Face Detection

[12] Object detection is a branch of computer vision that concentrates on determining if a predefined set of objects are present in an image and if so where they are located. The objects' locations are normally determined by rectangular bounding boxes that surround each object.

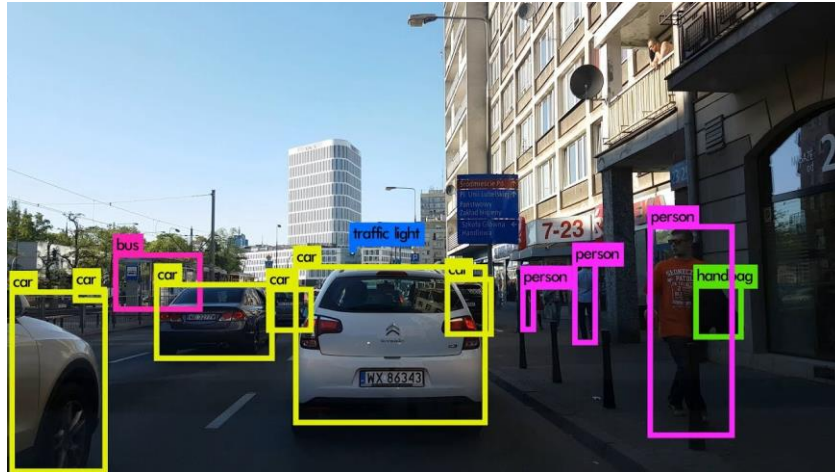


Figure 13. [71] Example of an object detection algorithm being applied to an image that contains various predefined classes. The predicted bounding boxes are overlaid in the image with their predicted class indicated by color and text.

Sean has two main objectives that stem from human detection. The first is to simply inform the system if there are any humans in the scene, and the second is to pinpoint where the humans are. Since the source of a human's voice stems from the face and that is the objective, human face detection along with human detection are approaches worth investigating.

An ideal algorithm for Sean would be to always identify where every human's face is in the scene, but there are several factors that make face detection a hard problem. There are a lot of fluctuation across human faces in a picture such as the person's pose, expression, and facial hair to name a few. There have been several different approaches studied to solve this problem.

#### 3.3.3.1.1 Knowledge-Based

A knowledge-based approach uses a set of predefined rules created based on the existing information of human faces. These rules may include expecting two eyes in a human face and identifying spatial relationships between the different structures within the face. However, with all the variation possible it is likely the set of rules will either be too strict or too general.

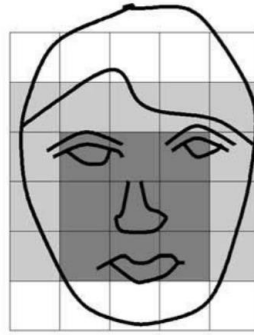


Figure 14. [14] A face template that was generated by human created rules. Yang and Huang used this weighted template in their 3 step knowledge-based approach that started with general rules and got more specific.

### 3.3.3.1.2 Hand-Crafted Feature-Based

A more robust method to human face detection is an approach that uses invariant features to detect faces. [15] With this method a human no longer has to develop a knowledge base, but rather identify a set of features common across human faces, such as edges, texture, and skin color. One of these most widely used methods is a Haar-like feature based approach. A Haar-like feature is generated by placing rectangular regions at a specific location in an image, summing up the pixel intensities in each region and calculating the difference between these sums. These differences divide the image into different subsections that become the image specific Haar features. [16] These are compared against the set of predetermined Harr features for the specific problem, in this case face detection.

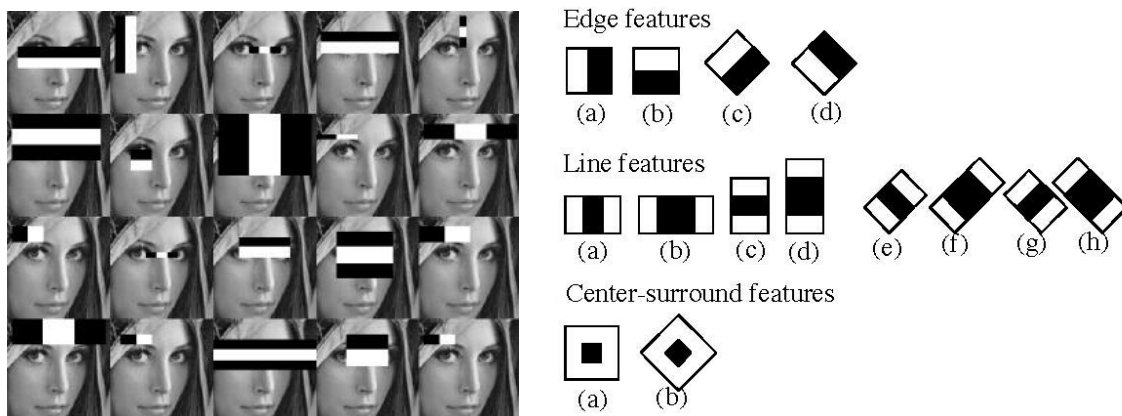


Figure 15. [15] An example of haar-like edge, line, and center-surround features applied to a face at different regions and scales.

Viola and James propose a robust method that is now widely used known as haar cascades. Their approach first transforms the image into a summation space that allows calculations to be performed very quickly. [17] Then a learning algorithm is used that selects a small number of critical visual features from a larger set and yields extremely efficient classifiers. Finally they combine increasingly more complex classifiers in a “cascade” which allows background regions of the image to be quickly dismissed with most of the time being spent on potential objects.

### 3.3.3.1.3 Deep Learning-Based

Today, deep learning approaches are on the cutting edge for solving computer vision problems. This is especially true for object detection. [73] Deep learning is a broad term for learning-based algorithms that use a cascade of artificial neural networks tuned by heaps of data to make decisions. [74] Artificial neural networks are inspired by the way neurons work in a human brain. A neuron in a brain is stimulated by sensory information which then activates connected neurons and the final output depends on the most activated path in the neural network. Artificial neural networks act in a similar way by layering individual “neurons” or nodes that take an input and combine it with weights associated with that neuron and propagate those results through the network. The most relevant neurons impact the final decision the most.

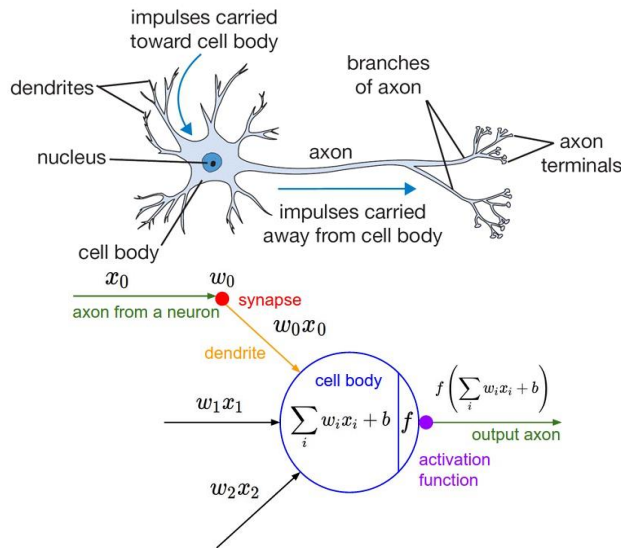


Figure 16. [72] Diagram of a human neuron showing how impulses are received and processed and a diagram of an artificial neuron illustrating how weights are multiplied with inputs, summed with a bias and passed through an activation function to become an output or the input of the next neuron.

Each individual node works in a fundamental way. Weights are associated with all the input branches to that node, and these weights are multiplied by the inputs. The summation of these multiplications are added to a bias value for the node that allows the function to shift. This new space is pushed through an non-linear activation function. This non-linearity is important because it allows the model to fit a real non-linear world. This can be visualized in Figure 16 above. [75]

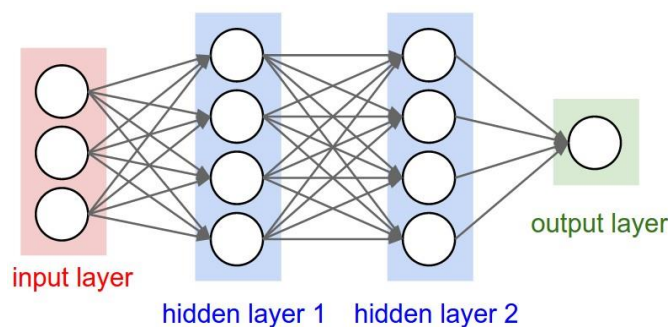


Figure 17. [72] Example of a fully connected neural network with 3 layers and an output layer. The hidden layers represent nodes with weights and biases.

One of the most common tasks tackled by neural networks is classification. Classification has the objective of deciding what class data belongs to out of a set of predefined classes. For this classification task the output result is normally transformed through a softmax layer which predicts the probability of each class being correct. The total of the probabilities sum to 1 and the highest value class can be thought of as the most confident. This softmax function is shown in Figure 18 below.

$$P(y=j | \theta^{(i)}) = \frac{e^{\theta^{(i)}}}{\sum_{k=0}^K e^{\theta_k^{(i)}}}$$

Softmax function

where  $\theta = w_0x_0 + w_1x_1 + \dots + w_nx_n = \sum_{i=0}^n w_ix_i = w^T x$

Figure 18. [7] The softmax function illustrated to show how the output data can be normalized from [0, 1] and sum to 1 to represent probability.

A regular way neural networks learn how to produce an expected result is through an algorithm known as backpropagation. Simply, backpropagation is learning through error. After a confidence is predicted as with softmax shown above, the network will make a decision based on its most confident answer. During backpropagation, a network is shown heaps of data in batches and makes a decision for each data point in each batch. Those decisions are compared to the correct output and an error or loss is calculated based on how wrong the decisions were. The weights and biases are then updated throughout the whole network based on the derivative of each weight or bias with respect to the loss function.

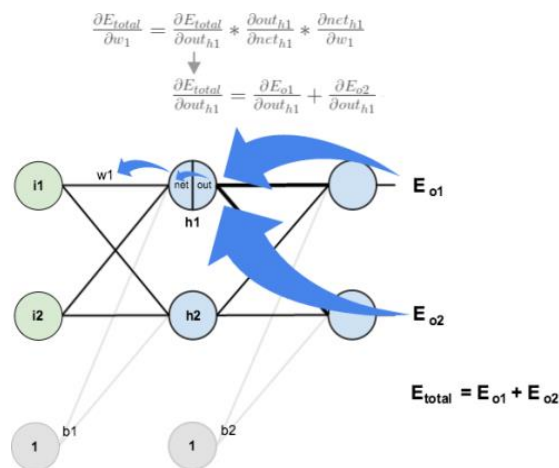


Figure 19. [77] An example showing how the total error can influence the weight updates of a single node. The partial derivatives of the weight with respect to the total error are calculated and multiplied by the weight to make the update.

After the network has been trained on a sufficient amount of data, it will saturate to its best possible performance and should be able to generalize well to data it has not seen before. If training goes on for too long, overfitting may become an issue. This occurs when the network becomes very specific to the training set and cannot generalize well to other data. If training is cut short, then the network may not fit the problem with high enough accuracy. These sequences of operations are the basis for deep learning.

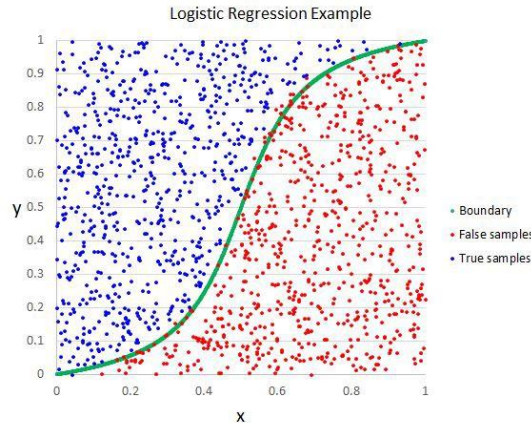


Figure 20. [78] An example of a model that fits a 2-class problem well using logistic regression - 2-class probability based learning.

Today many computer vision problems are solved with deep learning using Convolutional Neural Networks (CNNs). These CNNs work extremely well in the image domain because they put the image through several transformations that allow the network to see different features at various scales. During a convolution operation a kernel is slid over an input space and an element-wise multiplication is performed between the kernel weights and input values to form the activation space. [29]

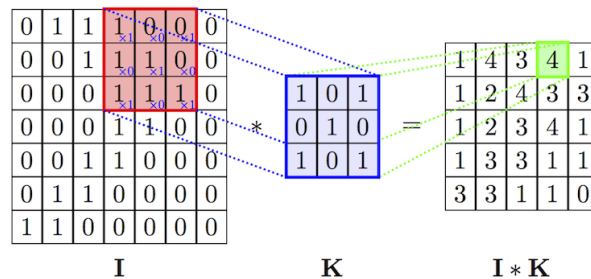


Figure 21. [30] A convolution operation with a kernel size of 3x3 being applied across an input space. It can be seen that at the current step in the convolution, the activation space in the 4th position would have a value of 4 due to the element-wise multiplication between the kernel weights and input values.

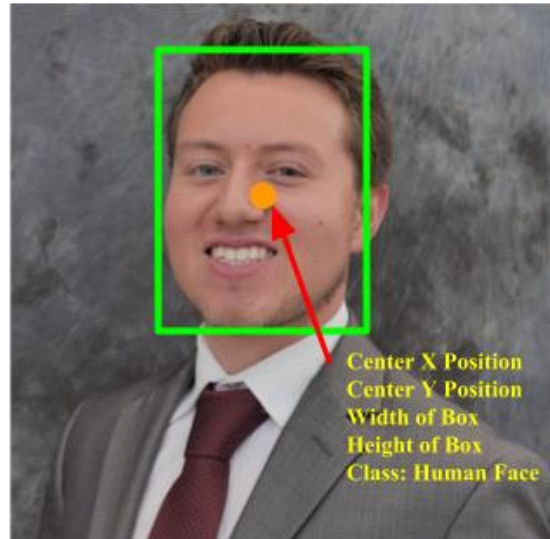
A CNN is designed similar to the basic neural networks described above, except that weights learned in a CNN are in the kernels and different sized convolution kernels are applied at each layer. A CNN architect chooses the number of filters and the size of the filters at each layer. CNNs have an advantage over regular artificial neural networks due to their ability to relate learned weights directly to image filters. They are also faster due to a significant reduction in the number of weights as compared to fully-connected neural networks. The most popular detectors



for human and human face detection use a deep learning approach with CNNs as their base. Several of these approaches are being investigated.

### 3.3.3.1.3.1 Detection With Deep Learning Neural Networks

As discussed in the previous section, deep learning based networks are efficient and effective for predicting classes after learning in a data driven way. However these networks can be architected to predict more than just classes. In the case of object detection, a network can be tuned to predict multiple bounding boxes per image that represent a specific class that the bounding box encompasses. This requires the input data that the network learns from to contain bounding box information as shown below.



*Figure 22. An example of an image input with its corresponding labeled data for a human face detection task. The labeled data is in the form of 5 scalar values: [Center X position, Center Y position, Width of box, Height of box, Class label]*

With labeled data like the kind in Figure 22 above, a network can be configured to learn the size, position, and class of bounding boxes using loss calculations from the center X position, center Y position, width of box, height of box, and class of box. There are several different variants of networks that attempt to solve this task, but most of these can be split between two categories. The first is to predict potential regions with objects and then classify those regions. The second is to predict the region and the class at the same time. Region-based Convolutional Neural Networks (RCNNs) are on the leading edge of the first option while You Only Look Once (YOLO) based networks are one of the top second options. [53]

#### 3.3.3.1.3.1.1 RCNN Based Detection

Given the task of object detection, a basic CNN can produce an unbounded number of outputs with no spatial constraints making learning objects of interest very difficult. One way to bound this problems is to extract a predefined number of regions and then classify them. This approach was demonstrated by Ross Girshick et al where they used a selective search algorithm that uses texture and color information to create the regions of interests (ROIs) and pass those into a classic CNN for classification. [54]

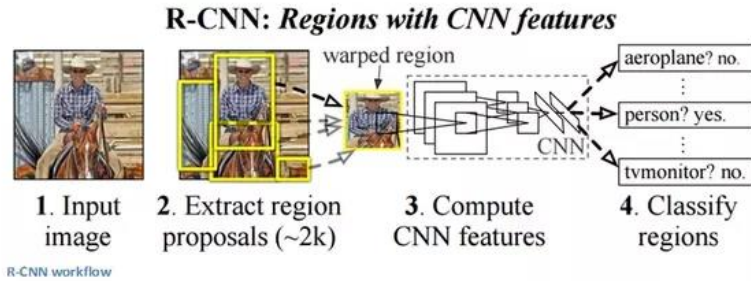


Figure 23. An example of a sample image being processed through an RCNN. Each ROI is extracted and passed independently through a CNN to classify the region.

However, this approach proved to be slow since all the ROIs of the image had to be passed through a CNN. A network that built off this work is known as Fast-RCNN. The main difference here is that instead of every ROI being fed into a CNN repeatedly, the entire image is passed only once through a base CNN that extracts image features. Then the ROIs which were found through a Selective Search Algorithm, are combined in a pooling layer which extracts their features and vectorizes them. Then this can be fed into a fully-connected layer, and a loss function as shown below. [55]

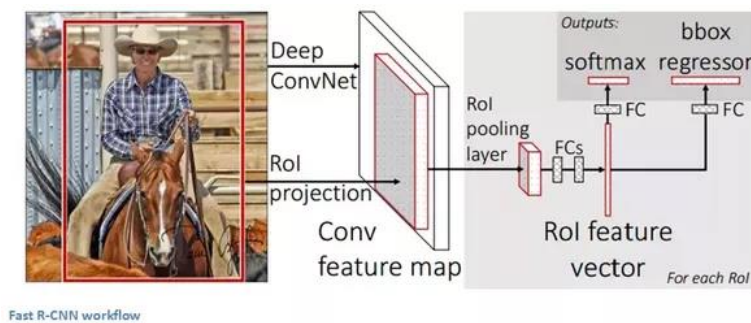


Figure 24. An example of a sample image being passed through a Fast-RCNN. Here it is shown that after each ROI is projected into the convolution space, they are vectorized together and an output can be quickly computed across the ROI feature vector.

The major slow down in Fast-RCNN is the selective search algorithm. Although powerful, it is computationally expensive and introduces a non-learning based algorithm into the network. These realizations lead into into Faster-RCNN. Instead of using a selective search algorithm to identify regions, Faster-RCNN introduces a region proposal network that operates on the features of the last convolution layer of the base CNN. [56]



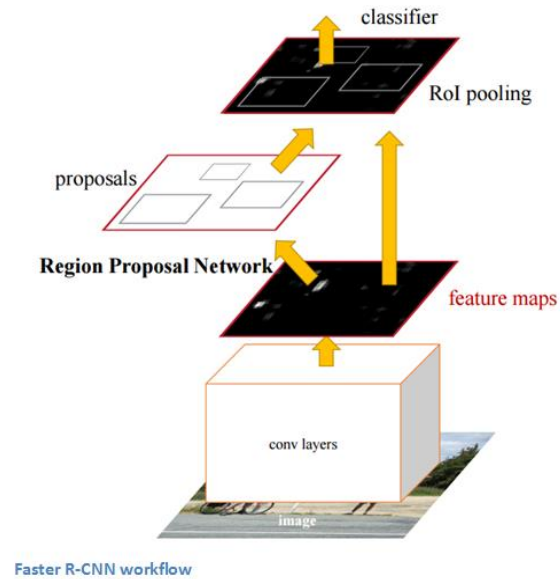


Figure 25. [55] An illustration of a Faster-RCNN network. It is show here how a selective search algorithm is no longer used and rather a region proposal network can predict bounding box locations.

Faster-RCNN's major contribution is eliminating the need for an outside algorithm to compute ROIs and can turn an RCNN-based network into an end-to-end learn network. This can be very powerful if a sufficient amount of data is available to guide the learning process. Passing images through CNNs on a GPU take a lot less time than performing search algorithms on a CPU.

### 3.3.3.1.3.1.2 YOLO Based Detection

YOLO is a CNN that simultaneously learns how to fit bounding boxes and what classes those boxes are. In order to get around the problem of an unconstrained number of boxes, YOLO breaks the image up into an  $S \times S$  grid. Each grid cell is responsible for predicting two bounding boxes along with class specific confidences for that cell. [59] YOLO's final output is a  $7 \times 7 \times 30$  structure which represents  $(7 \text{ cells} \times 7 \text{ cells}) \times (2 \text{ boxes} \times (\text{center X position} \times \text{center Y position} \times \text{box width} \times \text{box height} \times \text{confidence of box}) + 10 \text{ class specific confidences})$ . The 10 class specific confidences are related to the 10 classes the original YOLO architecture was detecting. This number will change as a function of how many classes the architecture is looking to detect. After all the bounding boxes are generated, a non-maximum suppression algorithm that takes into account box confidence score, grid cell class confidence, and overlap between boxes is applied to extract the most confident detections. An example of how an image is processed through YOLO is shown in Figure 26 below. An important takeaway from the YOLO architecture is each grid cell has the opportunity to see the image as a whole, and contextual information is encoded. This allows for the center position of the bounding box to be predicted accurately.

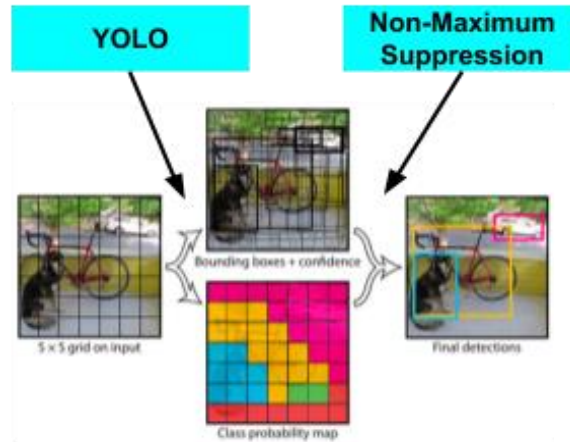


Figure 26. [60] An example of an input image being run through the YOLO algorithm with its bounding boxes and their confidences shown (bolder is more confident) along with the class probability map for each grid cell. The final output is shown after Non-Maximum Suppression and thresholding.

[60] YOLO has become one of the algorithms of choice for real-time object detection. This is because its architecture is based on a CNN with custom loss calculations to predict bounding boxes. The YOLO architecture can be seen in Figure 27 below.

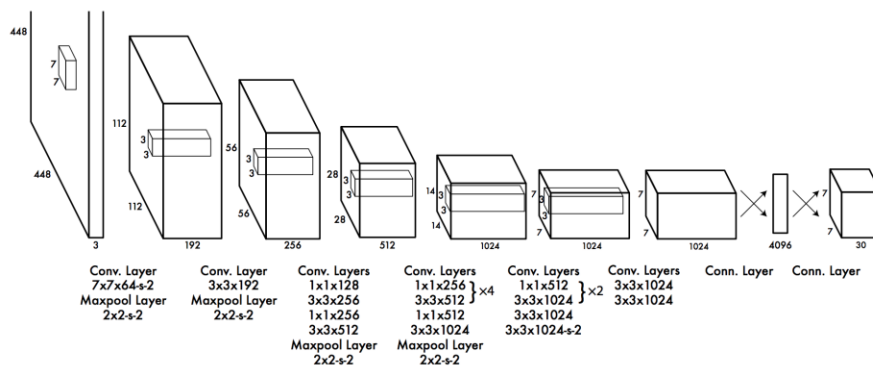


Figure 27. YOLO architecture diagram. This is a convolutional neural network that encodes information at different scales and creates an output that relates to the YOLO-defined grid and number of classes.

### 3.3.4 Mobile Application Development

In today's society mobile applications are used in almost every aspect of daily life, from controlling household appliances, to checking bank account information and even doing homework. Starting in the 1990s, the first applications were on handheld computers created by Psion, such as PDAs, and gave users access to things such as a word processor or a spreadsheet right at their fingertips. These early handheld computers used EPOC as their operating system and it allowed users to create their own apps through Open Programming Language (OPL). The next wave of innovation for these devices came with the development of the Palm Pilot. These devices provided the user with the ability to have third party apps, which were created using C or C++, and a touchscreen.

After that it was an industry wide race to make it easier and more efficient to develop mobile applications. Java Micro Edition came out on top and began to set the standard for a number of

devices that enabled the use of applications, not just mobile phones Java ME allowed for the setup of configurations that were then attached to a device profile. One of these profiles was the Mobile Information Device Profile, or the MIDP which was used for mobile phones by almost everyone.

#### **3.3.4.1 iOS Development**

Objective C is the main programming language for iOS and developers can use the XCode IDE to create their applications. XCode also includes an iOS simulator which is helpful for testing before exporting to a device.

The downside to creating an iOS app is that it means our system is only available, in its current state, to iPhone users. In terms of application development, it can only be done on a Mac computer which is limiting when it comes to updates and collaboration. Some other restrictions that Apple enforces are that the application cannot duplicate core iPhone features, the developer must pay an annual fee in order to publish the application and we would need an Apple certificate in order to install the application to our own devices.

#### **3.3.5 Wireless Communication**

In order for the user to easily operate Sean, it will connect to a mobile application on the user's phone wirelessly. This section will discuss the possible methods for wireless connectivity and which one will be most effective for all of the features that will be included in Sean.

##### **3.3.5.1 WiFi**

There are two ways to transmit information using WiFi. The first is Point Coordination Function which is rarely used. The second is Distributed Coordination Function which is very similar to ethernet conceptually and in terms of packet structure, but instead of through a wire, it transmits through the air. Due to the high error rate, among other disadvantages, DCF WiFi must be collision avoidance based rather than collision detection based. To combat these issues WiFi has certain requirements to ensure it is effective. The first requirement is positive acknowledgement, which means that before another packet is sent, there must be confirmation that the previous one was received. To address hidden substantiation, it also includes channel clearing. In order to do this the sender sends a Ready To Send, or RTS, message and the receiver sends back a Clear To Send, or CTS message. If both of these messages are sent it means the channel is clear. The last element included is channel reservation. This allow the sender and receiver to hold on to a channel for a few milliseconds after a packet is sent. This is done through the use of a Network Allocation Vector (NAV) within the packet which contains a number that represents the number of seconds for the channel to held. The last packet has an NAV of 0 which releases the channel after it is sent.

##### **3.3.5.2 Bluetooth®**

Bluetooth® was initially developed in 1994 by a Swedish cell phone maker in order to allow laptops to make calls through mobile phones. Since then it has been adapted to be used in many devices for various reasons. It operates as a short-wave, low power radio hookup that is always on. It can transmit across distances of 10 meters or less and its frequencies are in the 2.45 GHz range. It can also be used in a variety of devices such as headsets, printers, keyboards, and much

more. Also since it does not require a set-up file to install it, it makes using all of these devices between each other that much easier.

Bluetooth® has become one of the most common ways to eliminate a wired connection between devices whether it is used for music streaming, file transfer, or the real time sharing of information across devices. However, not all types of Bluetooth® support all types of media transmission. For example, some will not transmit stereo music. In order to make sure it will it needs Advanced Audio Distribution Profile (A2DP). Not only are there different components, but types as well. Starting from Bluetooth® v1.0 all the way up to v4.0, each version has its own improvements which are demonstrated in the table below.[63]

<b>Table 3 --Bluetooth® Version Improvements</b>	
<b>Bluetooth® Version</b>	<b>Specification</b>
Bluetooth® v1.0 to v1.08	Mandatory Bluetooth® hardware device and address
Bluetooth® v1.1	IEEE standard 802.15.1-2002
Bluetooth® v1.2	Faster connection
Bluetooth® v2.0+EDR	Enhanced data rate
Bluetooth® v2.1	Secure simple pairing
Bluetooth® v3.0	High Speed data transfer
Bluetooth® v4.0	Low energy consumption

Some of the common issues with other wireless standard are the need to have a clear line of sight between two devices and the inability for one device to communicate with two other devices at the same time. For example, one common wireless standard is Infrared (IrDA) which is used between televisions and their associated remotes. IrDA would not allow the remote to communicate with the television and a DVD player at the same time. Bluetooth® overcomes this issue by sending packets in bursts through radio waves, which allows one device to send packets to multiple devices at once. For example, someone could use their fitness tracker synced to their phone and play music from their phone through a Bluetooth® enabled speaker at the same time. For devices that are constantly communicating with each other, they will establish a piconet to allow for communication without interference. They will initiate spread spectrum frequency hopping which causes the devices to hop across 79 random frequencies within a specific range in unison approximately 1600 times per second. Since there is a minimal likelihood that another

device will be using the same frequency, this allows multiple piconets to function in the same building or area. This is sometimes call a master and slave relationship.

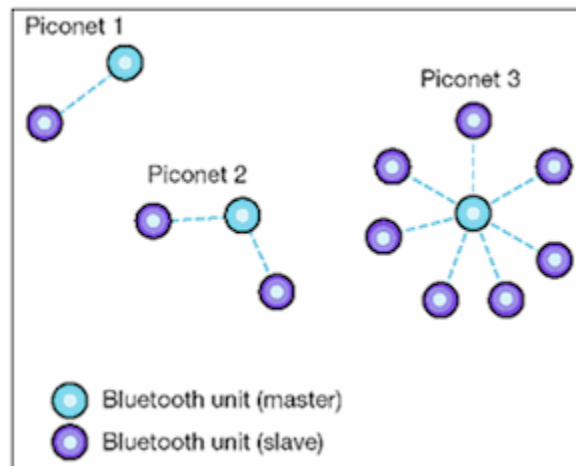


Figure 28. Diagram of a piconet

It also aids in the issue of syncing information across devices. People today use their cellphones almost as a personal computer so the consistency and ease of access of information is a necessity. Especially for people who run businesses using their phones, not having access to all of their information can create a large problem. Bluetooth® creates an easy way to sync devices and ensure all information is available across however many devices a person may possess.

### 3.3.5.3 Real Time Streaming Protocol

Real Time Streaming Protocol or RTSP is a network protocol used to stream multimedia such as video or audio between two endpoints. It is based on a proposal written by Anup Rao and Rob Lanphier in 1997. It was published as Remote Function Call 2326, or RFC, by the Multiparty Multimedia Session Control (MMUSIC). RTSP operates at an application level and transmits data in real time. It originally started as a method for users to stream videos directly from the internet instead of having to download them onto a device. It combines both TCP and UDP in to accomplish this. Since it communicates directly between the server and the device, it is very useful for data that needs to be time synchronized. When using RTSP the user's device will send a request to receive the options available through RTSP such as play or pause. Then the server will respond with the ones that can be used. Then the receiving device and the sending device will enter a back and forth conversation to ensure that the connection is properly set up. First, they will communicate with each other a description of the media, starting with a request from the receiving device. After the sending device responds with the description, they move on to the transport mechanism. Finally, the sending devices starts the stream by using a binary sequence, also known as a bit stream, according to the transport mentioned discussed between the devices.

### 3.3.6 3D Printing

Three Dimensional printing is a relatively new way to create parts and products but it has existed as an idea in someone's mind for a long time. In 1964 a science fiction author described what would later be a 3D printer and its basic functions. It was not until 1987 when a 3D printer was actually created and released to the public using a different process than the one used in today's printers. As it grew in popularity more technology and methods for printing were released but it

was still only used for creating prototypes in industry. 3D printing was established as an industrial manufacturing technology in 2009 when the ASTM Committee F42 published a document containing the standard terminology for it. This also coincided with a large price drop for 3D printers because the patents on one of its technologies expired. This allowed outside manufacturers to recreate the machines and sell them for around \$2000, which was more affordable for people outside of large corporations.

The basic idea of 3D printing is take a digital model of an item and printing layer by layer to create a physical three dimensional model. Some people also refer to this as Additive manufacturing. In order to create the model the 3D printer software takes the digital model and creates 2D slices of the entire thing. Then using G code, it sends the instructions to the printer to actually create the model. The rest of the process is dependent on the material being used in the printer. There are numerous free software that allow for users to create 3D models in order to print so that only adds to the advantages for using this method for small scale specific projects such as Sean.

The most common materials for 3D printing resemble plastic but metals are also used. Each material has different properties so it is important to define the use of a product to determine which properties are the most important to create a successful product. Depending on the kind of printer being used and the material, printing times can take up to 18 hours for one model. However, the unique thing about 3D printing is that when the model is complete it can be used immediately for whatever project it was printed for. There may be some slight polishing or filing required but other than that the product comes out almost perfectly complete.

#### **4 Related Standards and Design Constraints**

With the development of any new product, there must be a way to ensure it meets certain requirements that are established across the industry to make sure it will be beneficial. Standards accomplish this goal. Not only do they make the transfer of data easier, but they also aid in lowering production costs, ensure efficient development and later on aids in ensuring the needs of consumers will appropriately be met by the product. The Institute of Electrical and Electronics Engineers, or IEEE, is not only the world's leading standards developer but also provides the most information on standards, the impact they have and how they may be applied to new applications. Another important organization in standards development was started in 1916 which a number of engineering groups came together and decided to begin collaborating to enhance the work of one another.

Within IEEE there exists IEEE-SA which also develops global standards across a number of industries including, but not limited to, information technology, telecommunication, and energy. In order to maintain their integrity and spur on innovation IEEE-SA will conduct hundreds of ballots in order to vote on proposed standards. The process to develop varies greatly because there are a number of steps that must take place before the standard even gets to the voting stage. First the standard must be sponsored by an IEEE approved organization and that group will have the responsibility of supervising while it goes through the rest of the process. There are a number of professional groups that fall under the umbrella of IEEE so these groups tend to be the sponsors. Next, a Project Authorization Request must be submitted to the Standards Board and the New Standards Committee advises the Standards Board about whether or not they should

approve it. Following approval, a group develops and drafts the standards according to the IEEE Standards Style Manual. Then the standard is balloted, reviewed and it goes under a final vote.

There are various types of standards that factor into the development of a product. Some products may have to consider more than one type in order to properly meet its goal. This section will discuss the standards for the project, which include characteristics such as dimensions and performance. We must also consider product, performance, and design standards. It will also cover design constraints which address factors such as manufacturability, health, and safety.

#### **4.1 Related Standards**

The sections that follow discuss engineering standards that were considered and applied when choosing different components for our system. Many of the standards relating to this product required payments to obtain or had multiple versions for the same protocol. The following standards were determined to be the most important and accessible for the development of our product. These standards will ensure our design can be used and reconstructed in a universal way. Since several of our parts are connected to each other their communication must follow a common protocol.

##### **4.1.1 IEEE 802.15.1**

With the growing use of Bluetooth® to connect the numerous electronic devices integrated into our daily life, Bluetooth® consistency has become an issue for developers. In order to aid with the compatibility with multiple devices created by separate entities, Bluetooth® standards have become increasingly important. Bluetooth® SIG (special interest group) is the organization that is responsible for the publishing of Bluetooth® standards. IEEE 802.15.1 is the standard that specifically outlines what is considered a Bluetooth® system. It covers the distance at which it may operate, the frequency bands and arrangement of channels that enable successful connection, and what kind of devices are allowed to create these kinds of connections. Since Sean is to be used during conversation with others in a variety of environments, these factors are important. If someone is outside of the bounds of a Bluetooth® connection but wishes to have a discussion with the user, they must be aware of the distance at which the system will operate. For frequency, there could be possible interference from other systems so functioning within the allowed frequencies and channels must be considered to create a stable connection.

##### **4.1.2 IEEE 802.11**

Developed in 1985 by U.S. Federal Communications Commission, this is the most widely used standard around the world for wireless computer networking. It was originally meant to be used for cashier systems. This standard is used in many everyday devices such as laptops and printers and covers wireless data transmission for both the physical and data link layers. It aids in allowing these devices to access the internet and communicate without the use of wires. It was initially released in 1997 and has had many updates since then. At the core, 802.11 is a series of half duplex modulations that occur in the air. It uses collision avoidance to ensure that, when data is sent through bursts of packets, the channel is clear of other users who may be trying to transmit at the same time. The TX2 and the mobile application will be communicating through a wireless connection so the method of data transfer outlined within IEEE 802.11 must be considered to ensure there is a solid connection.

### 4.1.3 IEEE 297

The IEEE standard No. 297 is known as the Recommended Practice for Speech Quality Measurements standard. The intent of this standard was to alleviate the issue of subjective speech quality evaluation by writing a procedure to facilitate the choice of method by limiting researchers to a number of procedures that looked to be promising for future applications. In this document, a list of terms and their definitions used to relate to speech quality either objectively or subjectively are listed below:

1. Speech Signals
  - Refers to the human voice in acoustical form or its electrical equivalent.
2. Speech Quality
  - Defined as a characteristic of speech signal that can be assessed in an objective or a subjective manner. For the purposes of this project, the designers will attempt to keep their assessment objective while a target end-user that has agreed to be a part of testing will be assessing the speech quality in a subjective manner.
3. Speech Level
  - For the purpose of this project it will be measured subjectively unless the speech signal is an environment where it can be quantified in decibels.
4. Noise Level
  - For the purpose of this project it will be measured subjectively unless it is sampled and measured before being compared to any speech level.
5. Signal-to-Noise Ratio (SNR)
  - Defined as the difference between speech level and noise level in decibels. The noise level will refer to background noise and (limited) noise caused by hardware.
6. Listening Group
  - Defined as a group of listeners assembled for the purpose of speech quality testing. For the purpose of this project, this will refer to the end-user that will aid in the testing of Sean by providing feedback from a hearing-impaired standpoint.
7. Trained Listening Group
  - Defined as a group of listeners that understand the purpose of the test and respond properly to the test. For the purpose of the project, the trained listening group will be the designers of Sean; they will look for issues and inconsistencies in how the system will process noise and speech.

Of the methods presented for speech quality measurement, the one that has most of the attributes that will be used in this project is the Relative Preference Method. In this method, the quality of a speech signal will be compared to reference signals and put on a scale by considering how often the signals are preferred to other reference. This is a method employed in the deep learning practices implemented in software design of Sean.



#### 4.1.4 IEEE 830

IEEE standard 830 is known as the Recommended Practice for Software Requirements Specifications (SRS). This standard sets guidelines for software to be developed and provides approaches for developers and customers to exactly understand the requirements. The standard also provides a list of characteristics to consider when producing a good SRS. Sean will have two main software components. The frontend software that will be developed for the mobile application is one of these components. The software that controls the audio and visual algorithms and the integration of their outputs is the other one. All of Sean's software will be developed to be in compliance with IEEE 830 by taking into account the following considerations:

1. Nature
  - Sean's software will contain the system algorithms that make up the processing of the visual and audio data being constantly collected through Sean's sensors. These two modalities must communicate and interact, so there will be a software layer that aids this interaction. The user will interact with Sean through a User Interface software developed for a phone application. The software must communicate and function in real time in order for there to be very little delay in the audio processing.
2. Environment
  - Sean's software will act to perform transformations of the visual and audio data. It will also be a medium to display Sean's functionality and deliver the audio output to the user. It is not intended to be a standalone project without Sean's hardware.
3. Characteristics
  - The software requirements are correct, unambiguous, complete, consistent, ranked by importance and stability, verifiable, modifiable, and traceable. Sean will employ these important characteristics to ensure the development team can understand and modify the software where appropriate.
4. Joint Preparation
  - Every entity involved in Sean is involved at some capacity in the software development process. Customers and developers must agree on the specific requirements, so both parties fully understand what is to be developed and delivered.
5. SRS Evolution
  - Many times during software development, requirements can slightly change due to new constraints and obstacles. Sean's requirements, although firm, are allowed to evolve to make sure the system can be delivered as a whole with room for unanticipated changes.
6. Prototyping
  - While developing the software, incremental prototyping will occur to make sure the software is performing as expected. It would be unwise to develop all the software to first begin testing. This is

especially true in Sean with all the different software and hardware components that must communicate.

7. Embedding Design

- Sean is an embedded project. There for the software must be embedded as well. Developing for our TX2 platform will ensure Sean has embedded software as well as developing for a mobile application. Software will be intermediately tested to sure Sean is meeting its embedded requirements.

8. Embedding Project Requirements

- Consideration is being made for project requirements such as cost, schedules, and procedures. Below a detailed analysis of our budget and schedule is provided.

## 4.2 Design Constraints

The following sections discuss the design constraints that were considered during the duration of this project as well as how they related to project requirements, resources, primary end user's needs, and other factors. These constraints exist to realize the feasibility of a design and improve society and quality of life of those who will use/interact with what is designed. These constraints include and are not limited to the following:

- Economic and Time
- Manufacturability
- Sustainability
- Environmental
- Health & Safety
- Social Political
- Ethical

### 4.2.1 Economic and Time

This project is sponsored by Lockheed Martin and has been given a budget of \$2000. This budget was given with the consideration of extra funding needs for testing, possible replacement of parts and purchase of software needed to complete the project. The goal of the project is to create a low-cost solution for the large and extremely high value market of people with hearing disabilities, therefore the goal is to spend as little money as possible to create the product. This is possible due to the fact that not only should Sean be more effective and helpful than current hearing aids on the market, but also more affordable. The current hearing aid market is not consumer friendly. Hearing aids on the market can be upwards of \$2000 for just a single one so the goal is too greatly undercut this price. They also still at functionality complications even at the highest end. This is unfair to consumers because hearing is used in so many areas of our daily life. The fact that people have to pay extravagant amounts of money to still not be able to use one of their senses correctly is a prime example of an unbalanced market. The most expensive part of Sean is the Nvidia TX2 which comes in at \$600. When all the functions and processing become more streamlined it may be possible to even reduce the cost of the processor as well and that will allow the overall price to be even lower. All the other parts are less than \$200, and these costs may be able to be decreased as well if Sean was to be produced on a larger scale.

Along with an economic constraint there is a major time constraint included in this project. The planning, research, and design of this project will need to be conducted in Senior Design I from mid-January 2019 to the end of April 2019. In this time, all the algorithm, PCB, and packaging design will be completed and followed in the second phase of the project, Senior Design II. Senior Design II is reserved for prototyping, testing, and adjustments to design.

#### **4.2.2Manufacturability**

Sean is made up of several easily accessible parts through online vendors. This makes manufacturability easy because if someone wishes to build it there will be little to no issue obtaining to parts used. Even if the exact part cannot be obtained, most of the parts have one or more adequate alternatives that may be used instead. Most of these parts are already compatible or can be modified to make it compatible with the other components. This makes the product easy to manufacture whether on a large or a small scale. There also is the potential for a rapid manufacturing time. Senior Design takes place over two semesters. Sean will be built over the span of the second semester which is a twelve-week production time. With proper preparation, tools, and solidified testing procedures, this time could be reduced even further.

#### **4.2.3Sustainability**

Sean is meant to be a daily use device so sustainability is a key factor in its development. Starting with packaging, Sean will be enclosed by 3D printed parts that have been carefully designed to ensure that all the components will be protected and secured, while also creating an aesthetically pleasing design. Sean will be printed using ABS which is a sturdy material that is resistant to wear and tear. Since 3D printing is becoming more and more popular, it is easier to access the materials needed to print pieces needed to create products. Although the system is resistant to wear and tear, it may become necessary to reprint and place certain pieces, which can be easily done and will also prolong the life of the system.

#### **4.2.4Environmental, Social Political, and Ethical**

With environmental conservation being a prevalent issue in society and politics, great concern was taken to pick a material to print Sean's packaging with. The decision was primarily between PLA and ABS. PLA is more environmentally friendly as it is a corn-based material which makes it biodegradable, but it does take some time for it to completely decompose. ABS is not biodegradable nor is it a renewable as it is oil based, however, it is more flexible. There is tradeoff here that will ultimately decide what material is used for Sean's packaging: the decision to create an environmentally friendly product and the decision to have a more flexible plastic that is less likely to break under stress. The only other potential factor to consider is that if this is a product that is sold commercially it would be able to be sized down immensely and a stress fracture of the packaging would be drastically less likely to occur and thus would allow for the decision to choose a material that is biodegradable.

#### **4.2.5Health & Safety**

Since Sean is a product that could cater to users with a disability, health and safety is an important consideration for this product. One factor to consider is the weight of the system, especially if the user carries it around during their day to day activities. With a solid amount of weight in the system, it requires the user to be in an acceptable physical state to carry it for a long period of time. This could affect the health of the user because carrying around substantial

weight may negatively impact the user's body and spine, creating long term health complications. Another factor is heat. The TX2 gives off a considerable amount of heat so it is important that there are multiple methods in place to dissipate heat in a way that is safe for the user. Not only should it dissipate heat, it must release it in a direction that is not uncomfortable or dangerous for the user. Some heat will not be dissipated and may remain in the system. The user should not be able to be injured by this remaining heat although they may feel it.

## **5 Trade study**

This section conducts the trade studies for the processor, microphone array, camera, and material to 3D print Sean's packaging with. Of these components the most important is the processor as it will do all video processing, some audio processing and factor in largely with what type of material to 3D print with and the size of the packaging. For this reason, the processor will have the largest and most detailed trade study. The microphone array trade study was centered around having a high quality and high number of microphones, powerful onboard processing, built in algorithms, and compatibility with the chosen processor. The camera trade study was done to pick a camera that has a high resolution, electronic image stabilization (EIS), and compatible with the chosen processor. The 3D printing materials trade study was conducted to factor in durability, sturdiness, and heat resistance. Pricing heavily influenced all the trade studies and ultimately led to what items were compared with each other. All of these trade studies will be discussed in more detail in the following sections.

The trade study is a crucial step for the design of Sean given that motivations of this project are to improve on pre-existing devices. This same philosophy will be apparent throughout the rest of the design through this project; since the major components of the device are being purchased commercially, there is well documented information on a lot of the methods and algorithms being implemented in this project and just mistakes to learn from and methods to build upon and improve for future researchers to improve on and make more advancement in implementing deep learning practices into hearing aids. The information and conclusions made in the Research section were extremely valuable there was existing feedback on solutions that had started to implement features similar to what Sean will implement. The feedback gave direction to what algorithms should be used in what ways and thus what hardware would be needed to implement said algorithms.

### **5.1 Processor Requirements**

Making a decision on what processor will support Sean requires understanding the system requirements as a whole. The system will feature external cameras, external microphones, connection to a phone app, and powered by a power supply. All of these components need to be compatible with our chosen processor. The processor will have the responsibility of controlling the system inputs and outputs, running the computer vision algorithms, and either merging outputs from the digital signal processor or controlling the audio processing itself. Table 3 below shows Sean's most important requirements and what processor features are being considered to meet those requirements.

Table 4 -- Requirements vs Processor Features	
Requirement	Hardware Spec
10 fps CV algorithm processing rate	At least 180 GPU Cores
Audio sampling rate 8 kHz to 216 kHz	At least 4-core @ 1.5GHz CPU
Wireless connectivity	Bluetooth® 4.0 or WiFi enabled

Since Sean’s computer vision module stands to be the most computationally expensive, processors that are built for image processing will be highly considered. These specialized processors will have multiple Graphics Processing Unit (GPU) cores. A Central Processing Unit (CPU) has a few cores that are specialized for sequential serial processing. On the other hand, a GPU uses hundreds of smaller and more efficient cores for parallel processing [18].

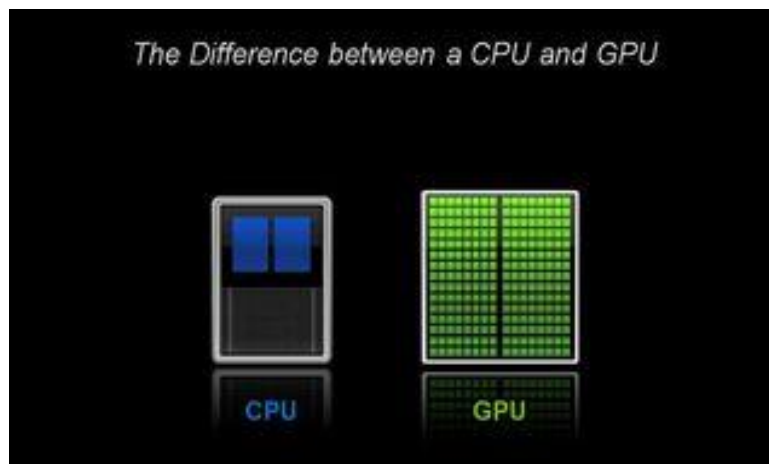


Figure 29. [19] Visual representation of a GPU vs a CPU cores.

### 5.1.1 Processor Trade Study

Below several relevant processors are traded to be the main computing power for Sean. GPU and CPU resources are the driving forces in this study. However, WiFi or Bluetooth® compatibility plays a major role so Sean can connect to a mobile phone app. This trade study also considers ease of integration based on community support.

#### 5.1.1.1 Jetson AGX Xavier Developer Kit [20]

The first processor being considered is the NVIDIA Jetson AGX Xavier Developer Kit. NVIDIA hardware is on the forefront of embedded computer vision processing. They produce processors with multi-core GPUs for streamlined processing and powerful multi-core CPUs. Specs for their cutting edge Jetson AGX Xavier Developer Kit are provided in the table below. This is shown in table 5 below.

Table 5 -- Jetson AGX Xavier Developer Kit Specs		
GPU: 512-core Volta GPU with Tensor Cores	Storage: 32GB eMMC 5.1	Audio Processing Engine (APE)[21]: <ul style="list-style-type: none"> <li>- 96 KB Audio RAM</li> <li>- Low latency voice processing</li> <li>- Multi-Channel IN/OUT</li> <li>- Multi-band Dynamic Range Compression (DRC)</li> <li>- Digital Audio Mixer: 10-in/5-out</li> <li>- Low latency sample rate conversion</li> </ul>
CPU: 8-core ARM v8.2 64-bit CPU, 8MB L2 + 4MB L3	Size: 105 mm x 105 mm	
Memory: 16GB 256-Bit LPDDR4x   137GB/s	Power: 10W / 15W / 30W	
Not Bluetooth® or WIFI enabled <ul style="list-style-type: none"> <li>- M.2 Key E slot allows addition of WiFi/Bluetooth®/Cell connectivity</li> </ul>	Price: \$650 with Academic Discount	

The Xavier poses well to go above and beyond all of our hard requirements with its 512-core Volta GPU and 8-core ARM v8.2 64-bit CPU. The Xavier’s onboard APE can streamline our audio processing algorithms as well. However, since it is NVIDIA’s newest processor, open source support is not widely available.

#### 5.1.1.2 Jetson TX2 Developer Kit [22]

NVIDIA’s Jetson TX2 is being evaluated due to its mainstream use of edge AI processing similar to the Xavier. The specs can be seen in table 6 below.

Table 6 -- Jetson TX2 Developer Kit Specs		
GPU: 256-core Pascal GPU	Storage: 32GB eMMC 5.1	Audio Processing Engine (APE)[23]
CPU: 2 Denver 64-bit CPUs + Quad-Core A57 Complex	Size: 170.18 x 171.45 mm	Memory: 8 GB L128 bit DDR4 Memory
Bluetooth® 4.1 and WiFi enabled	Power: 7.5W / 15W	Price: \$300 with academic discount

The TX2 stands to process audio and video with very little latency. It is slightly less powerful than the XAVIER but comes at significantly cheaper price. TX2 open source software is widely available including tutorials and demos.

### 5.1.1.3 Jetson TX1 Developer Kit [24]

The NVIDIA Jetson TX1 is also being investigated also due to it capabilities in edge AI processing. Table 7 below shows the Jetson TX1 processor specs.

Table 7 -- Jetson TX1 Developer Kit Specs		
GPU: 256-core Maxwell	Storage: 16GB eMMC 5.1	Audio Processing Engine (APE)[23]
CPU: ARM Cortex-A57 (quad-core)	Size: 170.18 x 171.45	Memory: 4GB 64-bit LPDDR4 Memory
Bluetooth® 4.0 and WiFi enabled	Power: 7.5W / 15W	Price: \$250 with academic discount

This processor is less computationally capable than the TX2 and the XAVIER. Even though it is a little bit cheaper, our budget allows us to purchase the more expensive variant with more processing power.

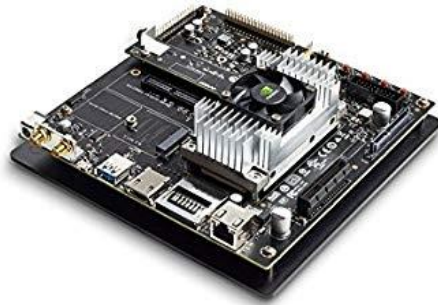
### 5.1.1.4 Raspberry Pi 3 [25]

The Raspberry Pi 3 has a reputation for being a very fast, lightweight, and low power processor. If it is capable of running computer vision deep learning algorithms then this could be a viable solution for fast, lightweight portable system. These specs can be seen in table 8 below.

Table 8 -- Raspberry Pi 3 Specs		
GPU: Broadcom VideoCore IV (Optimized for power)	Storage: microSD	No Audio Processing Engine
CPU: 4× ARM Cortex-A53, 1.2GHz	Size: 85.60 mm × 56.5 mm	Memory: 1GB LPDDR2 (900 MHz)
Bluetooth® and WiFi not enabled	Power: 1.2 W	Price: \$35

The Raspberry Pi 3 is a low power processor, but does not meet the computing requirements of the overall system.

### 5.1.1.5 Processor Trade Study Take Away



*Figure 30. [22] Image of Jetson TX2 Developer Kit.*

Sean's options were narrowed down between the TX2 and the XAVIER. Both of these processors go above requirement specifications. The TX1 is also competitive on this edge, but at only \$50 more the TX2 would be a better option. The TX2 is ultimately chosen as Sean's solution due to its significant price difference and interfacing capabilities with various cameras, microphones, and open source software.

## 5.2 Microphone Array

The decision to use an pre-assembled microphone array with beamforming capabilities was influenced by the research done in Section 3. The Earbeamer Senior Design project noted in their results and conclusions that more processing power was needed to improve the processing overhead to efficiently implement the FIR filter, beamforming algorithm, and noise cancellation algorithm. This would lower the latency between real-time sound and the processed sound sent to the user's headphones. Along with an improvement in latency, the circular microphone array configuration is ideal for amplifying sound at mouth level and having a professionally manufactured array reduces the possibility of have a low audio quality due to damages or inconsistencies in assembly by the group. Instead, at the end of this project, we can comment on future improvements to be made to the microphone array and its implementation in this kind of application while focusing on merging visual and audio spaces to benefit hearing impaired users.

There were several pre-assembled microphone arrays available for purchase: the specifications, features, lead times, and microphone specifications shown in Table 9 below were all defining factors in the deciding which part to choose. The microphone specifications were all in the same range with the exception of the PS3 Eye. Further research was done on it, but ended inconclusive as there was no datasheet that was found to reference to really consider. All but the PS3 Eye and the MATRIX Voice would have to be shipped internationally, which is extremely non-ideal given that the is the most crucial part of the system. From the specifications detailed in Section 2.3, the MATRIX Voice meets all of these with added bonus of 2 day lead time and FPGA as the processing unit. With all of this considered, the MATRIX Voice was found to be the most suitable microphone array that met the required specifications.



Table 9 – Microphone Array Options								
Chip	# Mics	SNR	THD	Noise Cancellation	AEC	BF	DoA	Lead Time
MATRIX Voice	8	62.6 dB	<1% @ 100 dB SPL <5% @ 115 dB SPL	✓	✓	✓	✓	2 days
Respeaker Mic Array v2.0*	4	61 dB	<1% @ 100 dB SPL <2% @ 115 dB SPL <10% @ 120 dB SPL	✓	✓	✓	✓	20-30 days
Respeaker Core v2.0	6	55 dB	0.46% @ 94 dB SPL	✓	✓	✓	✓	2 days
PS3 Eye	4	90 dB	<i>Unknown</i>	✓	✓			3-5 days
UMA-8 v2*	7	65 dB	1.6% @ 120 dB SPL	✓	✓	✓		20-30 days

\*Ships internationally

### 5.2.1 MATRIX Voice

The MATRIX Voice pictured in Figure 22 below is composed of an FPGA, 8 digital MEMS microphones, and 18 RGBW LEDs. The FPGA is a Xilinx Spartan® 6 XC6SLX9 which have low processing overhead in order to quickly process audio compute the complex voice algorithms. The microphones are MP34DB02 microphones and their specifications are denoted in Table 2. Seven of the microphones are aligned all around the edges in a uniform circle and the last is located in the middle of the array. The Everloop, the 18 LEDs around the edges, though not necessary will serve well for making Sean even more user friendly by having it all of the LEDs light up a certain color to denote what state it is in.

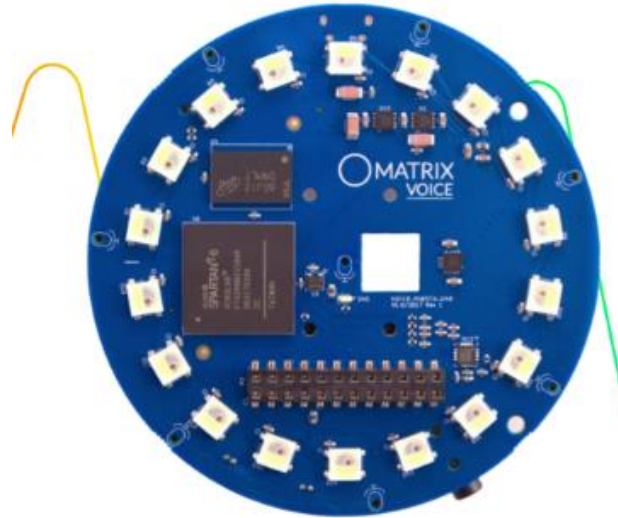


Figure 31. [14] Image of the top view Matrix Voice.

The board with the array has its own processing power aboard the PCB in the form of an FPGA Xilinx Spartan® 6 XC6SLX9 with audio processing algorithms available to be customized and improved by the user. This allows for the group to not waste time building a microphone array, which is a very well documented system at this time, and instead focus on improving the audio processing algorithms already in place. The Matrix comes with a Hardware Abstraction Layer (HAL) that need to be installed on the microphone array's FPGA; it contains all of the algorithms including beamforming, DOA, and an FIR filter. These are the algorithms that will be customized to fit the needs and objectives of Sean.

A major factor in choosing the Matrix Voice was that it had 8 microphones. Our specification early on detailed that Sean will need eight to twenty microphones. The reason for this was that since a heavy duty processor specialized for deep learning capabilities would be used and the camera needed for this kind of application would have a high resolution, then a high resolution audio space would also be needed which means more microphones. The resolution of both spaces needs as close as possible with the tools and parts available.

Sean will use the DOA estimating capabilities to calculate the direction of the source of the sound and then use the delay-and-sum beamforming algorithm to amplify and attenuate sources and sounds appropriately. The audio mixing will be dependent on the classification of the environment from the visual side of the system.

Since the microphone array is pre-assembled, the PCB will be designed to allow for the Matrix Voice to be powered through the processor and to allow for communication to be bidirectional between the microphone array and processor.

### 5.3 Cameras

Three different cameras are being evaluated to be the optical sensor for Sean. Cameras with a high resolution and wide field of view (FOV) are preferred to give Sean the greatest opportunity to image humans. All the following cameras are compatible with Sean's chosen processor: Jetson TX2 Development Kit. Table 10, 11 and 12 below go into the specs of each considered camera.

The best camera for Sean will have a high resolution and large FOV at an appropriate price point.

Table 10 – e-CAM132_TX2 Specs		
Resolution: 13MP (4192 x 3120)	Power consumption Max: 1.22W, Min: 0.79W	1/3.2” Optical form factor AR1335 CMOS Image sensor
Operating Voltage: 5V +/- 5%, Current – 335mA (Max)	Board Weight Camera module with Adaptor board – 3.5 Grams Base board – 13.5 Grams Coaxial cable – 4 Grams	Pixel size:1.1 μm x1.1 μm FOV:74.4°(D), 60.2°(H), 46.4°(V) Focal Length: 3.81 mm
Operating Temperature Range : -30°C to +70°C	Base board 75 mm x 40. Mm Adaptor board: 32 mm x 20 mm	Price: \$284

Table 11 – e-CAM20_CUTX2 – 2MP Specs		
Resolution: 2MP (1928x1088)	Power consumption Max: 0.87W, Min: 0.67W	½.7” Optical form factor AR1335 CMOS Image sensor
Operating Voltage: 5V +/- 5%, Current – 175mA (Max)	Board Weight Camera module with Adaptor board – 3.5 Grams Base board – 13.5 Grams Coaxial cable – 4 Grams	Pixel size: 3.0 μm x 3.0 μm FOV:128°(D), 103°(H), 70°(V) Focal Length: 2.8 mm
Operating Temperature Range : -30°C to +85°C	Base board (L x W) 75.03 mm x 40.18 mm x 25.6 mm	Price: \$149

Table 12 -- LI-JETSON-KIT-IMX477-X Specs		
Resolution: (4056x3040)	Price: \$399	Sony Diagonal 7.857 mm (Type 1/2.3) CMOS Image Sensor IMX477
Operating Voltage: 5V +/- 5%, Current - 335mA (Max)	Weight: 58g	Pixel size:1.55 μm x1.55 μm FOV:96.5°(D), 80°(H), 61.5°(V) Focal Length: 5 mm

All of these cameras can potentially be used, but before one is chosen an evaluation of how humans at our expected ranges would appear in the pixel space must be understood. This evaluation is completed in the following section.

### 5.3.1 Camera Pixel Evaluation

The average range of height and width of a human is outlined below.

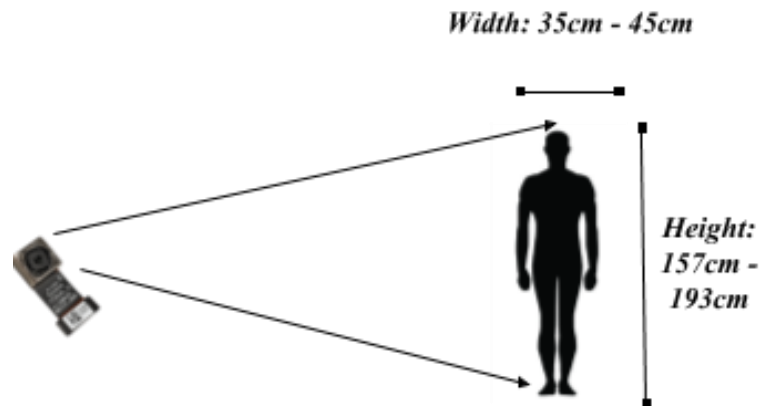


Figure 32. [16] FOV of camera compared with average Human Dimensions.

The following calculation is used to estimate the height of an object in a pixel space:

$$\text{Object Height}(\text{pixels}) = \frac{\text{Focal Length}(\text{mm}) \times \text{Real Height}(\text{mm}) \times \text{Image Height}(\text{pixels})}{\text{Distance to Object}(\text{mm}) \times \text{Sensor Height}(\text{mm})}$$

The table below describes how each camera would view the range of humans at our shortest and longest distant requirements.

Table 13 -- Camera Requirements				
Camera	Human Height: 157cm (5ft 1.8in) Human Width: 35cm (1ft 1.7in) Distance from Camera: 30 cm (1ft)	Human Height: 157cm(5ft 1.8in) Human Width: 35cm (1ft 1.7in) Distance from Camera: 305 cm (10ft)	Human Height: 193cm(6ft 4in) Human Width: 50cm (1ft 7.7in) Distance from Camera: 30 cm (1ft)	Human Height: 193cm(6ft 4in) Human Width: 50cm (1ft 7.7in) Distance from Camera: 305 cm (10ft)
e-CAM132_TX2	18126 pixels (17% of human) 4041 pixels	1783 pixels 398 pixels	22282 pixels (14% of human) 5772 pixels	2192 pixels 568 pixels
e-CAM20_CUT X2	4739 pixels (22% of human) 1089 pixels	466 pixels 107 pixels	5826 pixels (19% of human) 1555 pixels	573 pixels 153 pixels
LI-JETSON-KIT-IMX477-X	16881 pixels (18% of human) 3763 pixels	1660 pixels 370 pixels	20753 pixels (15% of human) 5376 pixels	2041 pixels 529 pixels

[81] Average person is 7.5 heads tall  $1/8 = 12.5\%$

All of these cameras would at least be able to view the largest person's head at the shortest distance requirement since the average head is 12.5% of a body and at the shortest distance requirement Sean would be able to see at least 17% of the largest person. This would allow Sean to always perform human face detection. So, the chosen camera will be the e-CAM132\_TX2 due to its resolution and autofocus capabilities shown in Figure 34 below.

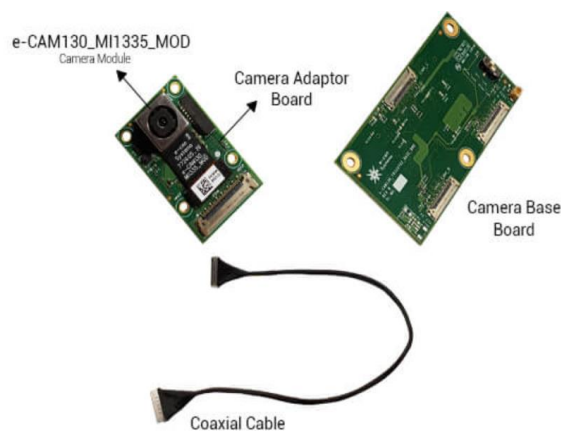


Figure 33. An Image of the e-CAM130\_MI1335\_MOD, the camera adaptor board, the coaxial cable, and the TX2 compatible camera baseboard that come with this camera selection

## 5.4 3D Printing

Due to the various sizes and shapes of the components of Sean, 3D printing was the best option to create a packaging that fit the pieces but also to give the product a complete and finished look. 3D printing allows someone to make any shape or object as long as they can create a digital model of it and get the desired material. The key attributes of the material used to make Sean will be durability, sturdiness, heat resistance and price.

Sean will consist of a number of sensitive devices such as a processor, microphone array and a camera, all which can be tampered with or destroyed if dropped or bumped in an incorrect way. Therefore the material must be resistant to bending or cracking that may expose the sensitive components. It also must be able to resist the normal wear and tear of a person using it throughout their day to day activities so it has to be durable otherwise it does not live up to its purpose. Heat resistance is an extremely important factor because there are multiple components that have the potential to give off an excessive amount of heat, namely the processor and the power supply. If the device overheats it could cause it to malfunction, melt a component or even injure the user who may be wearing the device. While the packaging will be designed to aid in heat dissipation it must also be heat resistant to prevent any other issues. Lastly, one of the main benefits of 3D printing is its low cost. Due to the desire to keep the price for Sean low, having low cost packaging is important for development. 3D printing materials are relatively cheap especially because there is little to no adjustment needed after the parts are printed.



*Figure 34. An Image of ABS printing material before it is fed into a 3D printer.*

The main materials being considered are Acrylonitrile Butadiene Styrene (ABS), Polylactic Acid (PLA), and Nylon. ABS is on the low end of the spectrum in terms of cost for 3D printing so it will be a good option for prototyping, under the assumption that we will have to reprint parts after trial and error. It also has a high heat resistance of 95-110 degrees Celsius, as well as high durability when compared to other 3D printing materials. It has a strength of up to 40 MPa is also resistant to wear and tear which will help Sean truly be a daily use product. PLA is similar to ABS in that it is also low cost and durable. It is also known to give a high dimensional accuracy when it comes to models which is important for a product such as Sean that has so many interconnected parts. However, it does not have as high of a heat resistance, it can only withstand 45-60 degrees Celsius. The last option is Nylon. While on the higher end in terms of price, it is also one of the strongest materials being considered with a strength of 85 MPa. It also has a very high impact and abrasion resistance. Even though it is tough it also provides some flexibility which could be helpful for a product that is designed to be wearable.

## 6 Design

Sean's design was guided by understanding the limitations of previous work and researching components that can together can form a functional high-performing system. Audio quality and implementation time are important factors that were considered and led to the decision of the MATRIX Voice as the hardware for the audio portion of this project. Previous research indicated that adding a visual component in this application will help solve problems related to the complex issue of much audio intention variation per environment; thus, a processor trade study was conducted to compare specifications and which would best meet the visual, audio, and communication needs of Sean. Ultimately, the NVIDIA Jetson TX2 Developer Kit was selected to be the main processor and integration platform. There was a consensus among existing products' results that a mobile app functioning as the user interface for the product was instrumental in allowing for user input, control of the device, and personalization of settings in a practical way. Sean will communicate with an iOS app to be developed and act as the mediator between Sean and the user's earbuds/headphones. The packaging will be 3D-printed to be a wearable device to allow users to pick their own earbuds/headphones according to what is most comfortable for them. Sean will be able to be used either sitting on a flat surface near eye level or strapped to the users chest to allow it to be a dynamic system.

In Figure 35 below, it can be seen that the yellow block--all of the components that make up Sean--shows that the camera, NVIDIA Jetson TX2, microphone array with the FPGA, and Sean App are all major components of the system. The green and blue blocks--the audio and visual integrated system (AVIS)--will be enclosed in a single unit of packaging while the app is a medium through which the user of Sean can directly interact with the hardware of the device. The flow of the diagram follows the stream of information through the camera and microphones all the way to the user's headphones:

1. Visual input is recorded with the camera while audio input is simultaneously being recorded with the microphones.
2. The camera base board allows communication between the camera and TX2 while performing on board camera control calculations.
3. The Matrix Voice will perform the bulk of the audio processing algorithms with assistance from the NVIDIA Jetson TX2, will perform the video processing algorithms in parallel. The visual and audio spaces will be merged on the NVIDIA Jetson TX2 and the audio and visual information will be output to the Sean App.
4. The Sean App will receive the output and display the video output as a live stream on the mobile device and relay the audio to the headphone in real-time.

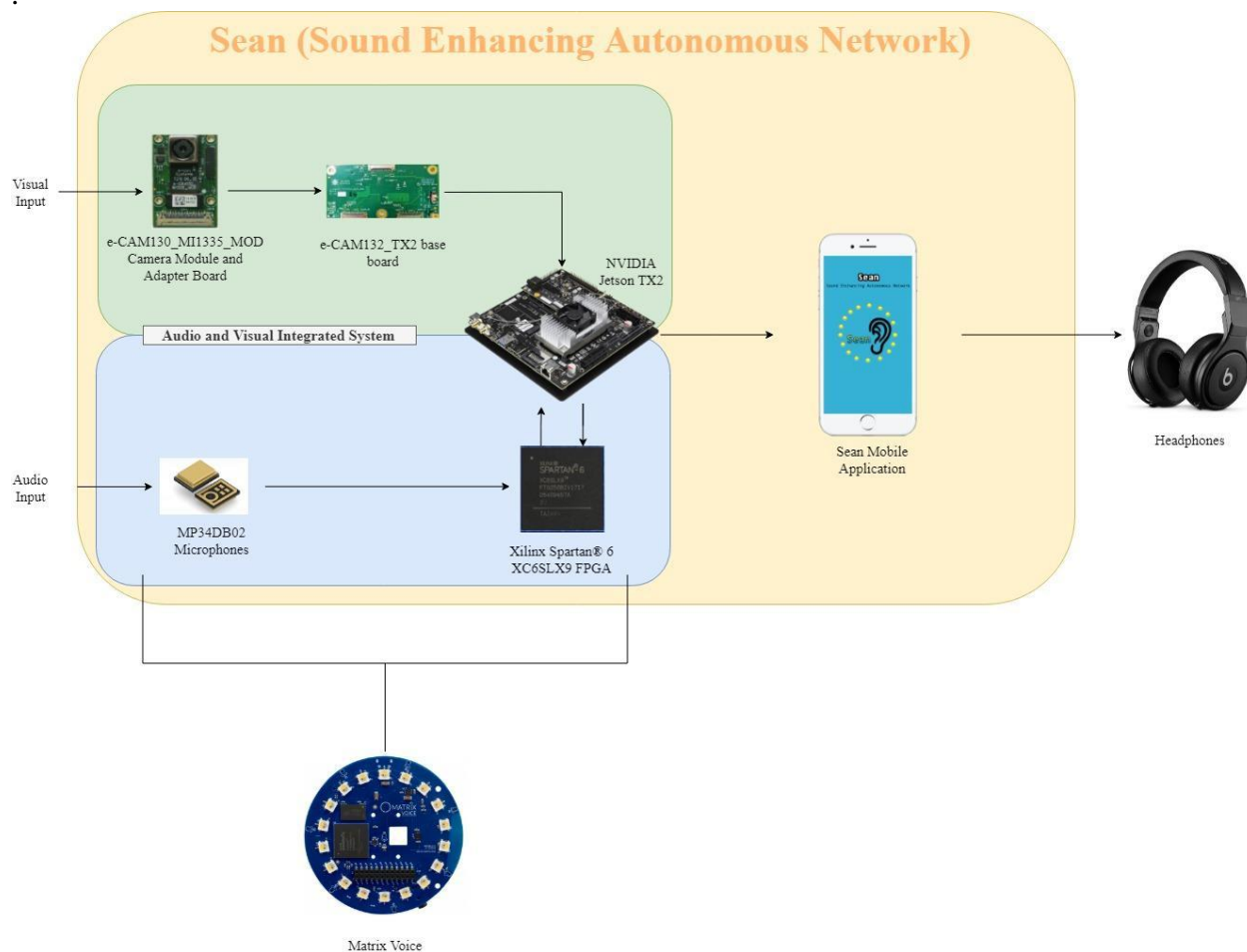


Figure 35. High level diagram of Sean system recording input and processing for audio and video output.

The following sections will discuss all of the components and systems mentioned above in detail and how they will be implemented to meet Sean’s objectives.

### 6.1 Hardware Design

Sean’s hardware design takes advantage of many already developed sub-systems each optimized to perform specific tasks. Each component is strategically connected as a module in Sean to solve lower level problems. These lower level solutions are combined to produce a system capable of processing multi-domain data in parallel while maintaining a real-time output to the user. The Matrix voice is a self-contained hardware component that contains the microphone array, an 18-LED ring, and an FPGA for audio processing. Before this component can be connected to the TX2 developer board, we must ensure the pins meet the expected voltage requirements. Sean’s PCB, will regulate voltage and act as a communication medium between the Matrix voice and the TX2. The first set of sound enhancing algorithms act to eliminate noise from hardware components, and these algorithms are going to be processed on the FPGA onboard the matrix voice to streamline this process. The FPGA will also be responsible for performing beamforming tasks and separate signals that can then be combined with the visual results on the TX2 module. The GPIO header on the TX2 developer board is directly connected to the TX2 module. Communication between the FPGA and the TX2 module will take place



though this header with the PCB as the middle component. The camera communicates with the TX2 through MIPI CSI2 connections. The camera carrier board is made specially for the TX2 so communication can occur very quickly. Wireless communication occurs through the Wireless connector assembly. Here external wireless antennas are connected to enable wireless communication. A mobile smart phone will connect to our physical system by communicating wirelessly. All of these hardware components will be communicating with the TX2, because the TX2 module will be doing a lot of the heavy duty processing. The GPU cores will be responsible for the Computer Vision face detection algorithms. The hex-core CPU will be responsible for performing an analysis on the audio signals with input from the face detection results. This hardware layout is ideal for real-time processing and is illustrated in Figure 36 below.

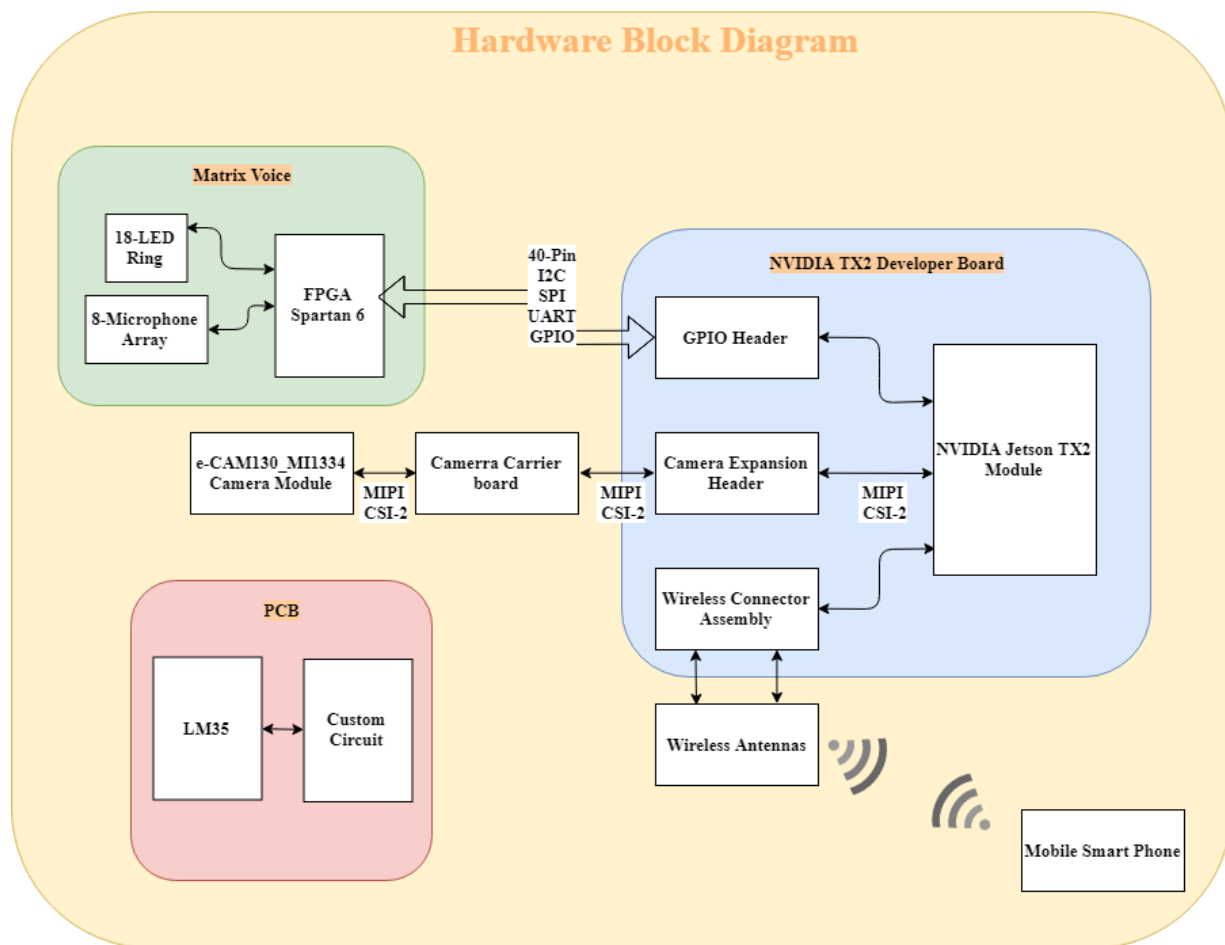


Figure 36. Hardware block diagram of Sean's major hardware components and communication systems and protocols.

### 6.1.1 e-CAM130\_MI133 module

The chosen camera for Sean is the e-CAM130\_MI133 module. This camera will capture the video data that will be sent to Sean. It will be attached to the camera carrier board which has on-board processing that includes camera controls such as, brightness, contrast, saturation, and the auto-focus functionality. This camera was chosen based on the trade study above. Using this very high resolution camera, Sean will be able to see the largest expected human face at the closest distance requirement. This camera module with its high resolution will also be able to have

enough pixels on target of humans far away which is a very important factor in face detection algorithms. This camera was also chosen based on its ease of integration with the TX2 module and developer board. The camera and TX2 can communicate via a coaxial cable with a MIPSI CSI-2 connection. This ensures an latency will not be caused in just grabbing frames from the camera. The camera module and adapter board integrate with the camera carrier board specifically made for the Jetson TX2 development kit.

### **6.1.2 Camera Carrier Board**

The camera carrier board employs the functionality of hosting low level software to have the camera send its data to the TX2. The goal of the camera/camera board system is to send the most pristine video signal to the TX2 so Sean's algorithms can process video without any artifacting, aliasing, delay, or other forms of badly processed video feed that can affect Sean's algorithm performance. This chosen camera system provided by E-Con Systems has been vetted and tested allowing for integration to occur easily. Having a quick-to-set-up, high performing camera solution will allow the visual system development to be mainly focused producing high performing human face detection algorithms and aligning those detections with audio-based detections rather than developing a custom camera solution.

### **6.1.3 Matrix Voice**

The Matrix Voice will take input with the 8 microphone circular array and do the onboard processing with the FPGA. Matrix provides user with a Hardware Abstraction Layer (HAL) on Github to configure the FPGA with. This comes preset with acoustic echo cancellation, noise suppression, an FIR filter, beamforming, and DOA algorithms that will be modified and improved as seen necessary for the software design of Sean. The GPIO pins that were designed specifically for use with the Raspberry will still be usable for communication with the NVIDIA Jetson TX2 but will need an intermediate piece of hardware between the two that will allow for communication between the two components without damaging the microphone array by supplying too much current to it.

#### **6.1.3.1 Microphone Array**

The microphone array geometry is optimal for the objectives of this project. Since the audio Sean is primarily concerned with is human voices and the Matrix Voice will be oriented perpendicular to the ground when it is operating, it will allow for the microphones to beamform in a manner at which the focal pattern of the beam will more converge more accurately on sources now that the microphones will source localize in a 3D space. The array is a circle of 7 evenly spaced MEMS microphones in a circular array with one last microphone in the center. The microphone in the center should improve the accuracy of the DOA and source localization algorithms by having a sensor in the middle to provide information about the location of any sources that lie within the maxima of the focal point.

The microphone array board also has a set of 18 LEDS called the Everloop that will be used in to help the user identify what mode the device is operating in: if the loop is all around green then Sean is operating without any issues, if the loop is yellow then Sean is the training mode where it is being configured by the user in training mode, and if the loop is red, then there is some kind of system failure that Sean is experience. If a failure occurs, the user will be able to check the Project Operations sections for troubleshooting information

The microphones are MP34DB02 microphones manufactured by STMicroelectronics, one of the world's largest semiconductor companies. [67] The microphones are capacitive MEMS microphones and are recommended in applications for speech recognition, A/V eLearning devices, and gaming and virtual reality input devices. These microphones are on the higher end of the market because, relative to other microphones, the SNR, acoustic overload point, and the sensitivity values are more impressive.

### 6.1.3.2 FPGA

The FPGA is a Xilinx Spartan® 6 XC6SLX9 that can be configured with audio processing algorithms provided by Matrix. This was not initially what was considered when deciding on processors specifically for audio processing. Only when research was conducted on existing products and solutions was this a component in the design that became important. The Earbeamer project conclusions mentions that processing overhead was a big issue with their project and resulted in a poor latency in audio to the user. Even with the massive processing power the NVIDIA Jetson TX2 has, it was important to find a processor for the audio that could sustain itself and improve with the help of our main processor. This processor came mounted on the microphone array board and became one of the major deciding factors in choosing the microphone array board for this project. The Xilinx Spartan family is designed to have low static and dynamic power which is important for the system as it will operate on a battery--a limited power supply. It has 45 nm process optimized for low cost and power along with a hibernate-mode for zero power.

### 6.1.4 PCB

The printed circuit board (PCB) to be implemented in the design will serve as temperature sensor of the system for the user to be able to monitor. Since there is a lot of hardware on different boards, it is crucial to monitor the inside of the packaging to ensure there is no damage of hardware or potential risks to the user. This will be done using an IC sensor as they are accurate within a 0.5°C and has a linear output voltage proportional to the Celsius temperature. The green LED indicates that the temperature within the packaging is still at a safe level while the red LED indicates that the system is beginning to run too hot.

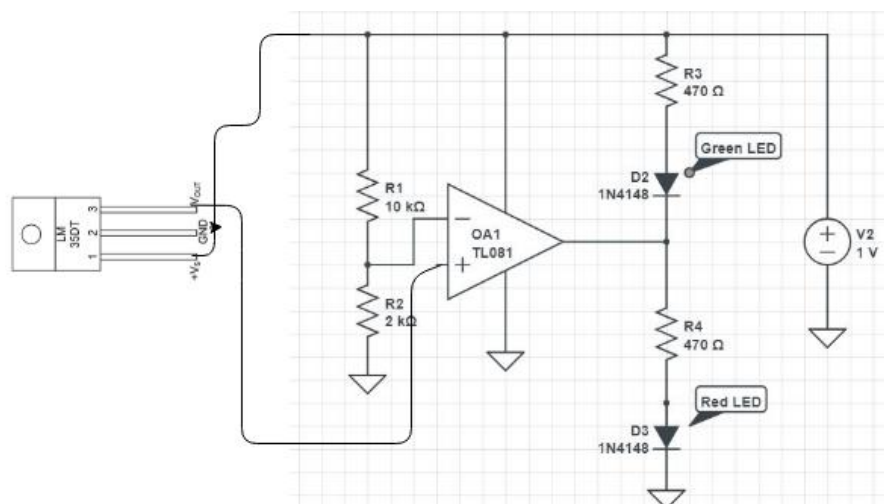


Figure 37. Schematic for PCB that will monitor the temperature inside the packaging of Sean to ensure cooling system is working as intended and to ensure that none of the hardware is getting too hot.

### 6.1.5 NVIDIA TX2

The NVIDIA Jetson TX2 Developer Kit will act as the main processing power in our design, and also as the platform for crucial integration of other hardware components. The developer kit has expansion header pins that can take the microphone array as an input via common pin connection. The camera carrier board can also easily connect via the Camera Expansion header. The TX2 has development kit has an HDMI display port and USB input and outputs that will allow for easy integration via controlling the TX2 with a keyboard and mouse and displaying to a monitor. In Figure 37 below a diagram provided by NVIDIA of the carrier board and it's useful components can be seen.

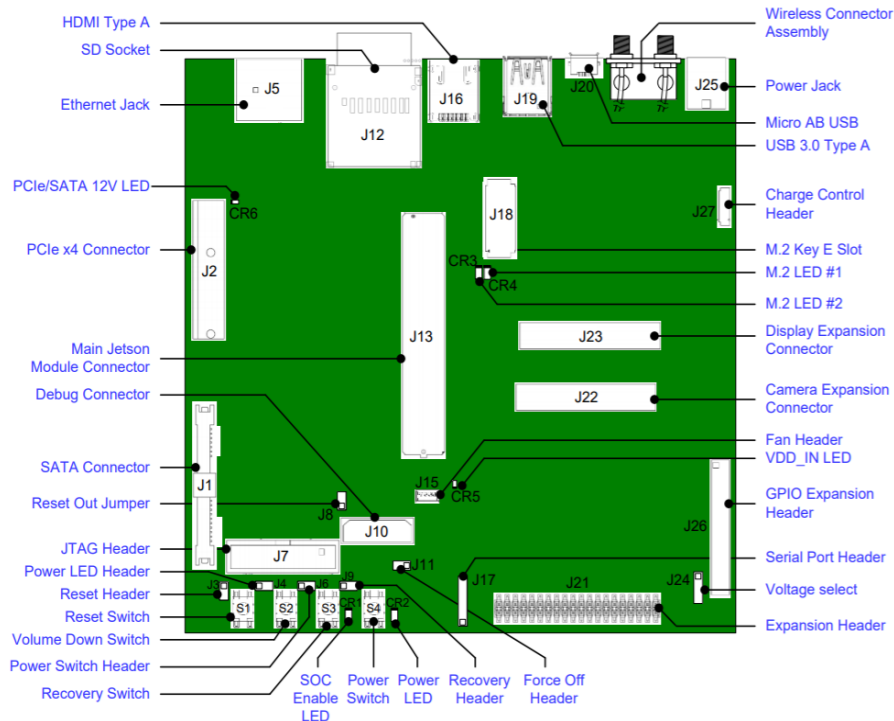


Figure 38. [66] The TX2 carrier board with its components that will allow for straight-forward test and integration are shown.

This developer kit, although bulky, allows the main design work for Sean to focus on performance, optimization of algorithms, and integration. Using the developer kit carrier board as is eliminates the need for creating a custom carrier board of a smaller size. The TX2 module itself is a powerful mini-computer that contains a 256-core GPU and a hex-core CPU. The GPU will render all the graphics along with processing the computer vision algorithms. The CPU will be concerned with the integration algorithms. Utilizing the resources across the TX2 is key to achieving real-time performance.

#### 6.1.5.1 GPU

The TX2 has a 256-core GPU module. This module will be responsible for processing much of the visual data that is fed through Sean. If a deep learning architecture ends up being the human face detection solution for Sean, then it will be applied across the GPU with CUDA optimized libraries. However, the GPU can also be utilized for a standard image processing technique to detect human faces. The technique Sean would be employing in this case would be a Haar

cascade method described in the research above. Utilizing the GPU for these vision tasks has a lot of potential to speed up overall processing. Again, the focus of Sean will be real time processing, especially on the audio side. A slow down of the whole system due to the visual components is not acceptable, and using the GPU's full potential will help to ensure Sean meets this requirement. A 256-core GPU will give enough processing power to process our frames. If Sean does encounter a slowdown, a risk mitigation plan will be put in place which lowers the resolution from 4k to 720p and reduces the frame rate to 15 frames per second rather than 30. We use these optimization techniques on the visual side to make sure Sean meets overall system requirements.

#### **6.1.5.2 CPU**

A hex-core CPU comes equipped on the Tx2 to handle sequential based processing tasks. All of our algorithms that aren't being run on the FPGA or GPU will take place on the CPU. This is a major component of the TX2 processor controlling not only custom algorithms but delegating and running overhead tasks to keep Sean performing. The CPU complex that is on the TX2 module combines a dual-core NVIDIA Denver 2 with a quad-core ARM Cortex-A57. The max-Q frequency of the ARM A57 CPUs is 1.2 GHz allowing the clock rate to be extremely fast. The CPU will be in control of running all the algorithms that merge the detections given from the visual and audio domains. This includes performing a confidence analysis to assess which signals are most likely a human engaged in conversation with the user. After these decisions are made, the CPU will also control amplifying and attenuating the signals accordingly. Then the final task of merging the separate signals into one signal will occur here. Finally the CPU will delegate the Wi-Fi or Bluetooth module to send the signal to the connected wireless device running the Sean phone app.

#### **6.1.5.3 WIFI Module/Bluetooth®**

The wireless module receives and sends signals via two connected antennas. During a study in the next development phase of Sean, it will be concluded if Bluetooth or Wi-Fi is the most appropriate method of communicating with the user's device. This wireless module will be responsible for sending live video feed of the camera to be displayed on the user's phone. The video feed is sent after it has been processed through a human face detection algorithm, and a confidence analysis has been conducted in coordination with the audio system to make sense of the different sources of sound for that current instance in time. The live video stream sent wirelessly will include symbology that represents the decisions that were made in these sections. Along with the video stream, the wireless module will be responsible for sending the final audio output to the user. This is the most important part since Sean must act as an audio enhancing system above everything else, and a protocol will be put into place that ensures audio is always being prioritized over a frame from the video. The user's voice that Sean will be calibrated with will be sent through this wireless communication. It will be sent from the mobile device to Sean in contrary to the mobile device receiving data from Sean. Sean will be in a calibration mode when the system expects to receive data wirelessly,

#### **6.1.6 User Hardware**

In order to operate Sean, the user must have access to a smartphone that can operate iOS applications and access the Apple App Store in order to initially install the application. The user must also have their own headphones that are compatible with their smartphone in order to

receive the audio output generated by Sean. This hardware must also be Wi-Fi and Bluetooth compatible that way it can connect wirelessly with Sean.

#### **6.1.6.1 Phone Hardware**

In order to operate properly, Sean must be accessible to be downloaded and then interacted with. Sean will be available to be downloaded from the Apple App Store so the smartphone but have WiFi or some other Internet capability. iOS devices contain a touch screen which is how the user will interact with the application. They also have physical buttons on the sides typically used for volume control. Sean will also include the use of these buttons for volume control. The smartphone must be able to receive audio input and output it to the user. In order to do this, it must have a part that receives audio. The most recent iOS devices do not have the standard headphone jack so some users may require the use of a lighting port adapter if they wish to use wired headphones with Sean.

#### **6.1.6.2 Headphones**

The user will have the ability to use their own headphones with Sean which allow for the most comfortable user experience for each person. Since different users may have different types of headphones, such as wired or wireless, the headphones will connect directly to the smartphone of the user. If using wired headphones the user will simply plug them into the headphone port on the device and ensure the application recognizes it. If the user chooses wireless headphones they will follow a similar process and connect the headphones directly to the smartphone. However they will have to use their phone Bluetooth connection settings outside of the application first and then return to the mobile application to continue using Sean. Because Sean is worn by the user, either option will still allow the user to have an optimum experience with the device.

### **6.2 Software Design**

In this section the algorithms and software that will be implemented on the audio and visual sides of the system will be discussed, as well as algorithms and techniques used to merge the information gathered from the audio and visual space into one. The audio processing algorithms include the two preliminary filters. One will remove noise in the signal (related to microphone and other hardware interference) to make it more precise. The other preliminary filter is designed to remove the user's voice. This section will then go into algorithms that use beamforming to localize human sources of sound, Voice Activity Detection to identify sources of sound as human voices, and amplify important sources of sound while attenuating what is not significant. These separated signals are then passed to signal to noise reduction algorithm based on the user selected environment. The visual processing algorithms will use Human Face Detection related algorithms to analyze the environment for humans that are potential sources of sound to compare results with the audio side for more accuracy in source sound amplification/attenuation. The two spaces are merged by converting both to a common angle space. This will help to better relate the values to be compared in the confidence analysis portion of the software diagram. Sean's machine learning based algorithms will create outputs as confidence intervals. Sean can provide itself with quantifiable values of the probability an event or in this case values that will predict the user's audio intention. The audio sub-system will calculate its own confidence intervals for each of the calculations it makes which relate to the likelihood the sound we are processing is coming from a human that is talking to the user. The visual sub-system will do the same. The values from the separate systems are then merged into one confidence value (per source provided

through beamforming) that will be amplified/attenuated based on the level of confidence Sean has given to the source of sound. With several human sources of sound, Sean will have to amplify/attenuate based on a tier system that is heavily confidence-based since the ideal case of one human per environment is not realistic. After this first round of audio mixing is completed, a second round is conducted based on what the user identified as their environment type (low noise, medium noise, high noise) to attenuate insignificant signals more/less or amplify important signals more/less base on background noise level. On the visual side, the faces being considered as sources are identified on the feed using boxes to show the user what sources are being considered in the audio. The audio is then merged back together and the final video output will have symbology that represents if Sean is detecting people, considering them to be talking, and what level of amplification or attenuation is being applied to those sources. The Sean phone application will relay all this information to the user. Sean's software will consist of th following components:

- Start Up Process
- User Input
- App Calibration
- Volume
- User Environment Classification
- Video Display
- Audio Capture
- Sound Enhancement Filter
- User Voice Filter
- Human Voice Detection
- User Voce Mixing
- Video Capture
- Human Face Detection
- Visual and Audio Space Alignment
- Confidence Analysis
- Confidence-Influenced Audio Mixing
- Environment-Influenced Audio Mixing
- Merger of Audio Signals
- Final Visual Output
- Final Audio Output

These modules can be seen in Figure 39 below. It is shown how data flows through each module. Int the following section it is described in detail how each module is functioning.

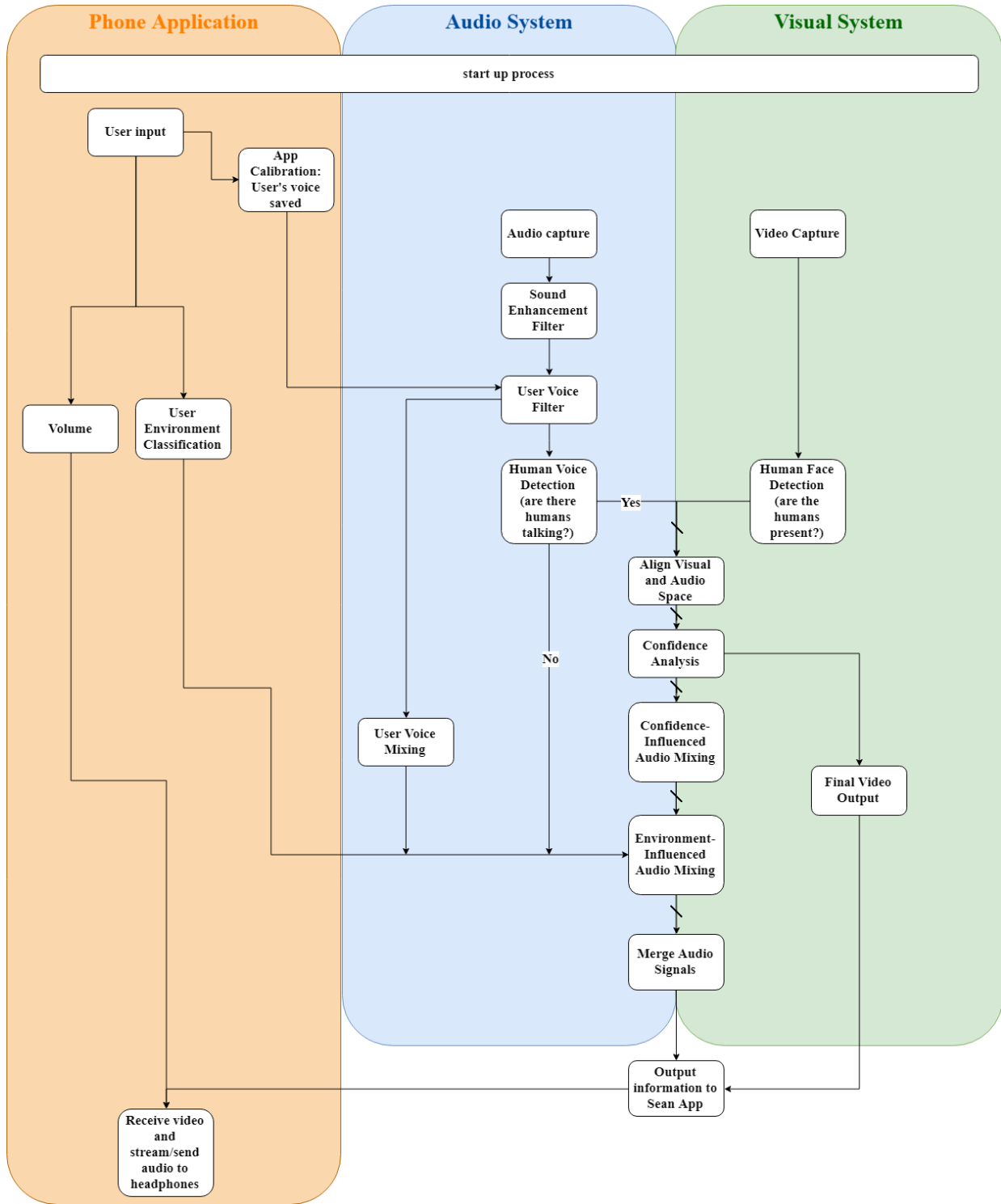


Figure 39. Sean's AVIS algorithm flow interacting with the Sean app.



### **6.2.1 Audio Subsystem**

The audio processing algorithms are going to primarily be focused on identifying human voices as sources of sound. Before any analysis of the environment is done on the audio side of the system, the audio will be run through two filters: a Sound Enhancement Filter and a User Voice Filter. The Sound Enhancement Filter will run several basic and well documented algorithms to perform acoustic echo cancellation, noise suppression, and an FIR filter to improve the clarity and precision of the audio before human voice processing is done. The User Voice Filter will be able to remove the user's voice for the signal to separate it from the audio that will be used to identify other sources sound. The reason for separating the user's voice and the rest of the signal is to take into account the way people speaking hear their own voice; people generally hear their own voice deeper than it actually is due to their skull vibrating from their vocal cords vibrating when they speak. Once the audio goes through these filters, it will be analyzed for any human voices present in the environment. This section of the algorithm, Voice Activity Detection, will calculate a value based on how confident the system is that there is a human present in the environment, along with values of average amplitude, distance, and their respective confidence values. All of the aforementioned filters and algorithms will be discussed in greater detail in the following sections.

#### **6.2.1.1 Audio Capture**

The Matrix Voice uses the Advanced Linux Sound Architecture (ALSA) framework to capture and mix audio. ALSA consists of a set of kernel driver and an application programming interface (API).

#### **6.2.1.2 Sound Enhancement Filter**

The Sound Enhancement Filter will be composed of acoustic echo cancellation and noise suppression algorithms as well as an FIR Filter. This section will not serve more than to clean up the audio signal before we run processing algorithms on it. The FIR filter and the noise suppression algorithms will help eliminate white noise, any noise in the audio that is there because of the hardware, or feedback. The acoustic echo cancellation algorithm will take into account the problem of reverberation and eliminate it.

#### **6.2.1.3 User Voice Filter**

The user voice filter will use pre-recorded snippet of the user voice to compare against the filtered audio signal to remove it completely from the main signal and distort it to make it sound as if the user speaking can clearly hear their own voice in the same manner they would without any headphones in their ear. The snippet would be recorded in the user's initial device configuration, the training mode. The user would be prompted to say sample sentences that could be used to reference against in the raw signal. Removing the signal from the main signal to be processed makes the human voice detection simpler in that there would be no added algorithm attempting to differentiate if one of the sources is the user speaking after beamforming occurs.

The user will take part in training the system to recognize their voice with the use of phonetic pangrams, a sentence that contains all letters of the alphabet and all 40 sounds of the English language. For instance, "that quick beige fox jumped in the air over each thin dog. Look out, I shout, for he's foiled you again using, creating chaos" is one of many phonetic pangrams that exists. The implementation of this portion of the audio system will require knowledge of

linguistics. It will require the use of phonemes, the smallest unit of sound in speech. The techniques being considered for this implementation are the Hidden Markov Model and Bayes Theorem. The Hidden Markov Model is a probabilistic model that allows for a prediction of sequences of a hidden state based on previous or current observations made and the Bayes Theorem describes the probability of an event based on past information that is related to said event. There are commonly used for speech recognition purposes and rely on predicting the probability that a phoneme will lead to another phoneme and so on to make the assumption that a person is speaking. For the Hidden Markov Model (HMM), the observations would be the signals, or sound, the user can hear and the hidden states would be the words spoken. With enough system training, the pattern of the user's voice will be compared with the audio coming into the algorithm and removed from that primary signal. Then the audio user's will be distorted deeper than what it actually is. This is because the vibrations in a person's skull from their vocal cords vibrating enhance the deeper, lower frequencies in their voice. This plus the added effect of hearing sound directly from the vocal cords to the eardrums creates a voice that people hear deeper than what they hear on a recording of themselves.

#### **6.2.1.4 Human Voice Detection**

The Human Voice Detection block will run a beamforming algorithm that will localize sources of sound and then isolate them and putting them temporarily into separate channels to be amplified or attenuated. At this point in time the localized sources of sound should exist in their own channels including the rest of the signal, the background noise. The channels containing the sources of sound will be checked individually for any human voice activity by using a Voice Detection algorithm. This will be translated into a value of confidence of how certain the system is that this source is a human. Beamforming should be the first algorithm in the human detection block since it will localize the major sources of sound to be individually checked with the voice detection algorithm. This is a much more accurate way of determining the distance the distance of the source, the amplitude of the source, and the likelihood that this source is a human. By just having the voice activity detection before the beamforming, we are only running the algorithm once, for one signal, leaving all signals identified in the environment to have the same confidence value for Voice Activity Detection. This would mess up leave the sources to be of similar value creating what might be an incorrect conclusion of what audio intention the user has if one source is louder or closer than the other by a little. If no sources are detected in the environment at all, the environment is not at all attenuated but avoids any kind of change in gain and is recombined with the user voice filter output (if there was any).

Sean will predict whether a signal is human or not by using techniques employed the Hidden Markov Model (HMM) and Bayes Theorem. A similar technique is used in the User Voice Filter algorithm. This algorithm will not only test a signal for one voice, but for many, and will potentially require more training for the system. However, there is much documentation telling of how a hybrid of HMM and neural networks (NN) is ideal for applications of their techniques especially for speech recognition because they balance each other well; HMM is not flexible and cannot handle all the phoneme variations well, but it if a great fit with the sequential nature of speech while NN is flexible and works extremely well with phonemes, however, does not handle large amounts of data well and is not great at handling problems that are sequential. Just as in the User Voice Folt

If there is some confidence ( $>0$ ) a value for distance from the user and amplitude of the signal will be stored and used to calculate confidences based on those values. The distance-related confidence will make the connection that the further away the source is from the user, the less likely it will be that this is a source the user wants to listen to. The amplitude-related confidence will make the connection that the quieter the source is, the less likely it will be that this is a source the user is interested in listening to. The converse of both of both of these statements is true as well. These confidence values are then relayed to the confidence analysis section to be merged with their visual confidence counterparts to calculate a total confidence value for every source in the system.

### **6.2.2 Visual Subsystem**

Video processing will play a key role in ensuring Sean operates with in real time with a high accuracy. The software implemented here will control all the operation that occurs on the frames captured through Sean's camera. This includes the video input stream controlling the capture frame rate, the resolution, and the field of view of the camera. The output video stream with appropriate symbology overlaid is also controlled by this subsystem. The critical component of a viable human face detection algorithm will also be implemented in the visual subsystem. Real-time processing is the focus consideration on the video side. Sean's video processing algorithms must add value to Sean's main goal of enhancing the sound of a human the user is having a conversation with and improving overall sound quality with little latency. If the video processing is slowing down the entire system Sean will be able to decrease the processing frame rate in order to not sacrifice speed.

#### **6.2.2.1 Video Capture**

The first processing the video stream will go through is a frame capture process. The camera adapter board and carrier board allow the video input to be sent to the TX2 via a MIPI CSI-2 interface. Low-level drivers that are preloaded with the camera module enable streamlined communication between the camera and the TX2's CPU. These frames can be grabbed at a specified rate and resolution with the driver interface. OpenCV will allow Sean to bring its processing on the frames.

#### **6.2.2.2 Human Face Detection**

Currently two main human face detection approaches are being evaluated. These two are a deep-learning based approach and a haar-cascades based approach. One of the advantages of deep learning is we can fully utilize the GPUs for parallel processing. However implementation is not as straightforward as the haar cascade method. Haar cascades comes preloaded in openCV and can easily be deployed. However if our deep learning based approach outperforms haar cascades to a significant degree it will be chosen as the final solution. TensorRT is a conversion tool provided by NVIDIA that allows for quick optimization of neural networks for streamlined GPU implementation. This process will also be tested when exploring the best human face detection method. YOLO has a tiny YOLO variant which involves using less convolution layers to greatly reduces the size of the network at the expensive of some performance. However, with the proper training on a binary problem such as human face detection it has the ability to perform extremely well and quickly on the TX2 module.

### 6.2.2.3 Final Video Output

Our final visual output will reflect what occurred during the combination of the audio space created during our beamforming technique and our visual space that was processed. Sean will display an output that shows what human face detections were extracted and if they were associated with a human voice detection. It will also display any source signals that were attenuated. This can be seen in Figure 39 below.

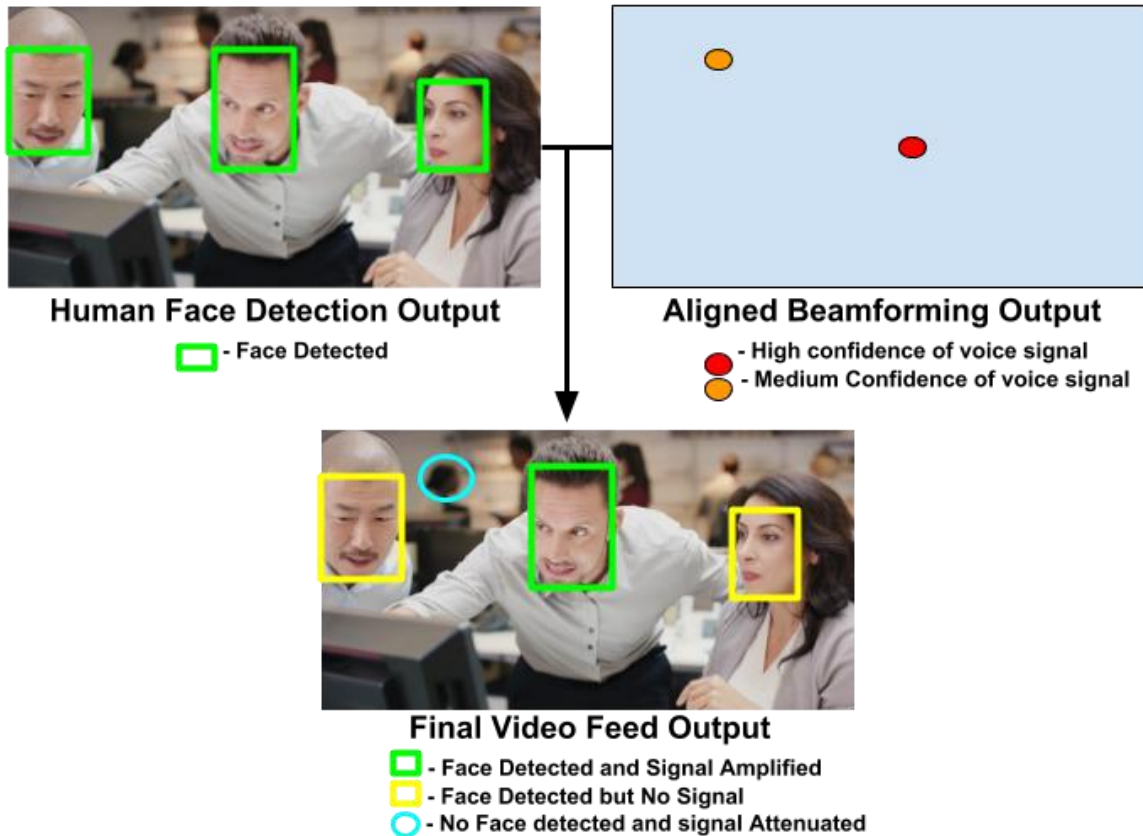


Figure 40. The figure above shows what kind of final output Sean will display on the phone application. This is formed by a preliminary face detection output and an audio beamforming aligned space output with confidence values.

### 6.2.3 Integration of Visual and Audio System

The integration of the visual and audio spaces is the most crucial design in the project: it will provide more accuracy in predicting targets of audio intention for the users. AVIS completely makes up the hardware for Sean and is the driving force behind the whole design. The audio and visual spaces are being integrated specifically for the reason that hearing aid users would be able to have the benefit of another pair of ears and eyes helping them process the world around them, impairment or not. This will start with the aligning the physical and audio spaces to make links between resolution and space (distance) between the two. This will allow the system to more quickly identify inconsistencies in algorithms or their implementation by coming up with fake or incorrect sources to include on the when mixing audio based on confidence level. With more accurate total confidences per source, the system organized them into a set of tiers called the Confidence-Influenced Source Signal Gain Tiers. This constrains our design and limits their

being one true target of audio intention for every user. This is not a completely inaccurate statement, but because, as mentioned before, the user can have various targets of audio intention. This case simplifies it for the purpose of seeing how much adding a visual space to a hearing-aid alternative improves the systems accuracy and versatility. After the signal gain tiers, if the user of Sean prompted a different change of environment than the default (low noise environment) then the signals that were attenuated are attenuated more, the signals with a gain of 1 stay the same, and the signal that was amplified get amplified more. The factor by which the signals get attenuated or amplified before being recombined is dependent on whether the user chose medium noise environment or low noise environment. This integration is a vital part of Sean and this allows us to use the novel techniques of face detection aid in the processing of audio.

### 6.2.3.1 Align Audio and Visual Space

Sean will combine outputs from the audio and visual space to make confidence-based decisions on if there are humans in front of the user and if the humans are talking. In the visual domain, through human face detection, Sean can make a decision on if there is a human and where he or she is located in the pixel space. In the audio domain, through beamforming various source signals can be abstracted and localized in an angle space relative to our microphones. In this section of Sean's processing, the goal is to align these spaces to determine which visual detections agree with the audio source abstractions. The first step is to transform the camera pixel space into an angle space. This can be done by using the camera's horizontal and vertical field of view and dividing by the number of pixels in the vertical and horizontal directions. This will provide a measure of degree per pixel for both the azimuth and elevation directions or also known as instantaneous field of view (IFOV). IFOV for our chosen camera, e-CAM132\_TX2, is shown in Figure 40 below.

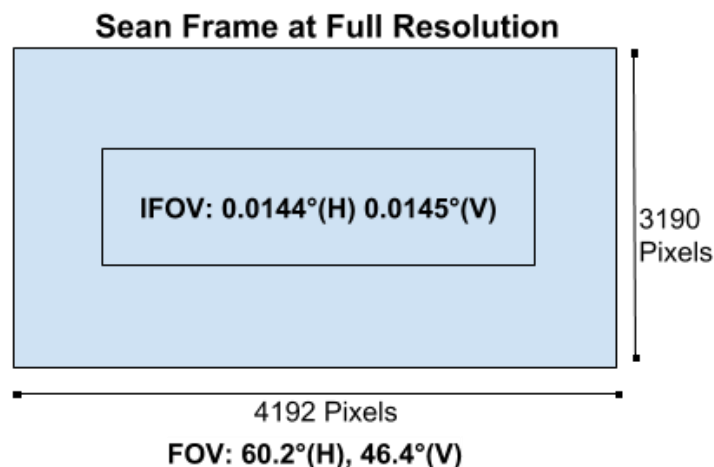


Figure 41. e-CAM132\_TX2's IFOV calculated from its FOV and number of pixels in both the horizontal and vertical directions.

The angle space of the microphone array will also be broken down into its IFOV measurement based on characterizing its FOV. Since the camera will be physically placed in the center of the microphone array the final calibration is not very intensive. After characterizing the FOV in the audio space, the centers of the camera and the direction of the microphone array can assume to be aligned and an overlap can be injected using the visual space as a reference. All detections that come out of the human face detection algorithms will then be converted from a cartesian

space to an angle space. All sources that are calculated from beamforming will then be re scaled with the visual space as the reference. This is shown in Figure 41 below.

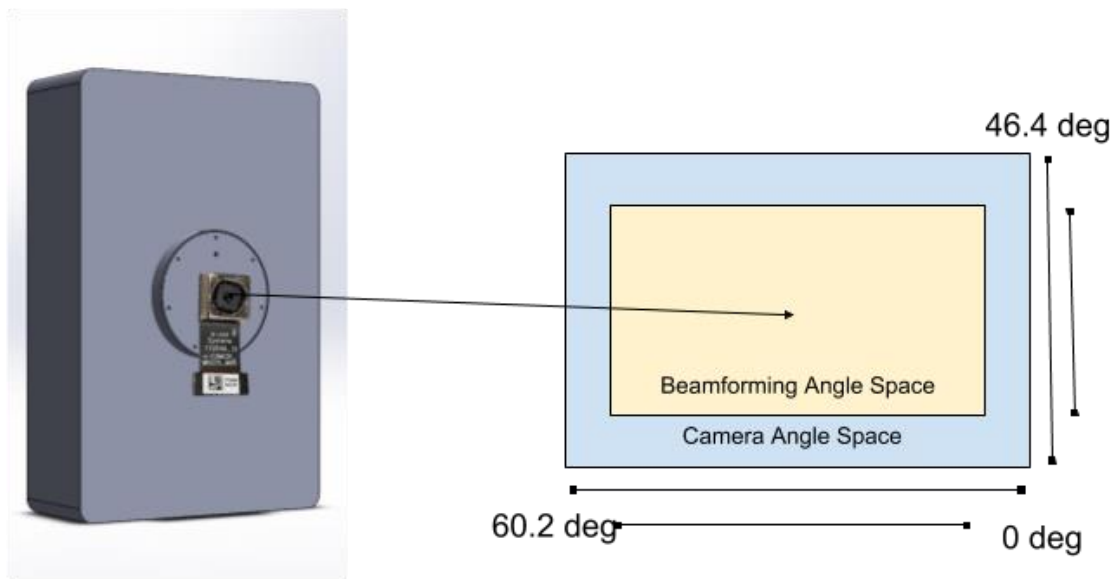


Figure 42. Illustration of how Sean's camera space and audios space generated during beamforming will align in order to compare detections.

After the audio space and visual space are registered Sean can now compare detections made in both domains. This is a crucial step to differentiate between different obvious source signals that are and aren't human voices, such as music playing in the background or ambient noise from a washing machine. Also, this step allows Sean to eliminate false detections that were thought to be human voice but are really just background noise. With the spaces registered the system can fully analyze the confidences that were produced from both domains.

### 6.2.3.2 Confidence Analysis

The confidences in the audio and visual subsystems will be based on calculated values from the human detection algorithms. In the audio subsystem, the Human Voice Detection block will calculate/measure a value for the distance of the source identified and the amplitude of the sound of that source and analyze how those would relate to knowing whether or not if this person is someone the user wants to listen to. The relationship would be translated to be a quantifiable relationship which would be the confidence value for that variable. The same would be true for the visual subsystem: the measurements/calculations for distance and look angle\* would be calculated and analyzed to create a quantifiable relationship between that measurement and knowing if the person that is identified is someone the user wants to listen to.

The audio confidence values seen in Table 14 below are a measure how confident Sean is in that there is a human talking and that this is a person the user cares about listening to based on the fact that this human source is X distance away, and that they are X dB loud. The actual values for distance and amplitude are calculated in the Human Voice Detection block from the audio subsystem. The Voice Activity Detection algorithm will come up with a confidence value zero to

one that will indicate how sure the system is that this source is a human talking. The beamforming algorithm will provide the system with a distance that we can translate into a confidence value zero to one that will tell the system how likely it is that this person is talking to the user based on how close they are standing to them. A mean amplitude will be recorded by the system and translated into a confidence value zero to one that tells us how likely this is a person talking to the user based on how loud they are. The confidences for the Voice Activity Detection, distance, and loudness will have a maximum of 1 (implying that the Sean is 100% certain that this is the source the user wants to be listening to).

However, the system cannot rely completely on distance to assume that the closer a person is the more likely it is they are talking to the user and it can rely on an amplitude-based confidence even less. These two factors do not hold true in all situations; for instance, sometimes the loudest person is much further away than who the user wants to be listening to. For this reason, we merge these confidences and weigh them differently in our equation to take into account for these kind of inaccuracies. The Voice Activity Detection confidence will be weighed the highest among the three confidence as it will be the most accurate confidence since Voice Detection Algorithm implementations have become improved in recent years.

Table 14--Audio Confidence Weight Values			
	Voice Activity Detection	Distance	Amplitude
Confidence Weight	<i>Most weight</i>	<i>Medium weight</i>	<i>Least weight</i>

The visual confidence values seen in Table 15 below are a measure how confident Sean is in that there is a human talking and that this is a person the user cares about listening to based on the fact that this human source is X distance away, and that they are X degrees within or from the user's look angle. The actual values for distance and look angle are calculated in the Human Face Detection block of the visual subsystem. The Human Face Detection algorithm will come up with a confidence value zero to one that will indicate how sure the system is that there is a human that could potentially be a source of sound. The algorithm will provide the system with a method to calculate distance that we can translate into a confidence value zero to one that will tell the system how likely it is that this person is talking to the user based on how close they are standing to them. A look angle\* will be recorded by the system and translated into a confidence value zero to one that tells us how likely this is a person talking to the user based on where they are oriented (angle-wise) from the user. The confidences for the Human Face Detection, distance, and loudness will have a maximum of 1 (implying that the Sean is 100% certain that this is the source the user wants to be listening to).

However, the system cannot rely completely on look angle to assume that the closer a person is to the center of view the more likely it is they are talking to the user and it can rely on a distance-based confidence even less. These two factors do not hold true in all situations; for instance, sometimes the person the user wants to hear could be to the right or left of them speaking. For this reason, we merge these confidences and weigh them differently in our equation to take into



account for these kind of inaccuracies. The Human Face Detection confidence will be weighed the highest among the three confidence as it will be the most accurate confidence. However, it is worth noting that Sean will only run the calculations for distance and look angle if the visual confidence value for Human Face Detection is 80% or more.

Table 15--Visual Confidence Weight Values			
	Human Face Detection	Distance	Look Angle*
Confidence Weight	<i>Most weight</i>	<i>Least weight</i>	<i>Medium weight</i>

\* *Variable not finalized*

The weight values of the confidences in Table 14 and 15 are not yet finalized as they will need to be played with an tested to account for inconsistencies and potential disagreements between the two spaces. An equation to merge those confidences into one has been developed and can be seen below. The total confidence should be merged such that it is a value from zero to one, which can be related into a percentage, which is an ideal way of describing the confidence. The visual confidence is weighted completely in the equation by as the audio space can tell when a human is speaking and the visual space can only tell when a human is present. Since the visual confidence values will improve the accuracy of the user’s audio intention predicted by Sean, the weight is still relatively high at a value of 0.8. The summation of the values is divided by 1.8 to account for the weight on the visual side and ensure that this value can be relayed into a percentage of confidence that this is a person the user wants to listen to.

$$\text{Total Confidence} = 0.54 * \text{Audio Confidence} + 0.46 * \text{Visual Confidence}$$

This merged confidence will tell the system how high the probability is that this is a person the user wants to listen to. Given that there is more than one source in the system, the sources must be amplified/attenuated according to what the value of the total confidence is of each source. A tier system was developed—and will be discussed in more detail in the next section—to better organize the sources to define what really qualifies a single source to be predicted as the source the user wants to her by Sean.

### 6.2.3.3 Confidence-Influenced Audio Mixing

The first round of audio mixing will be based on the total confidence values for each source. This Tier system was designed to take into account of how to amplify and attenuate the signals there is more than one source based on how confident the system is that this source is one the user of the system would want to hear. The tiers are a bit uneven in that there are two for attenuation, one for unity gain, and one for amplification. This is because the assumption--or rather constraint--for our system is that the user only wants to listen to one person at a time, thus only one source can be amplified at one time. The rest that meet the threshold of unity gain do not see any change in amplitude since the system was not as confident that they were the target of the user’s audio intention and those that could not meet the lower bound in Tier 3 are attenuated one level above the background noise attenuation. The values 50% and 75% were chosen a bit



arbitrarily and will be finalized with testing. The upper bound is on the high end of the spectrum because the system should be sure that it amplifies a source the user wants to listen to in order to really correct the amplification of all noise that is present in most commercial hearing aids. The lower bound is a bit on the lower spectrum to account for any inaccuracies or potential failures in the system that might make the total confidence of a source lower than it should be. The method in which the system decides to amplify or attenuate the signal of a source is seen in Table 16 below.

Table 16--Confidence-Influenced Audio Mixing				
	Tier 1	Tier 2	Tier 3	Tier 4
What sources go here?	Reserved for Background Noise	Sources below 50% total confidence	Sources above 50% and below 75% total confidence	Source with highest total confidence value above 75%
Gain	<i>Attenuation (&lt;1)</i>		<i>1</i>	<i>Amplification (&gt;1)</i>

#### 6.2.3.4 Environment-Influenced Audio Mixing

The second round of audio-mixing is prompted by a change of Environment Type settings by the user via the Sean app. The default setting of Sean will assume that it is in an ideal environment, a low noise environment. The other two options the user can pick are medium noise and high noise. Sean will work best in a low noise environment type as it will be much easier for the Voice Activity Detection to function and more accurately. In this mixing, we will derive factors to attenuate by and to amplify where is it appropriate on the tiers for the confidence-based audio mixing. The higher the environment SNR when the signal is what the user wants to hear, the higher the gap between what is attenuated and what is amplified to create a clear distinction between what was said and what is background noise.

#### 6.2.3.5 Final Audio Signal Output

After the two stages of audio mixing are complete all of the audio is merged into one signal to be sent to the Sean app. The signals that are recombined are the source signals that were identified and separated individually in different audio channels, the residual background noise that was left after removing the significant source signals, and the signal removed from the User Voice Filter before any audio was processed further.

### 6.3 Sean Mobile Application

This section will discuss the software components of the mobile applications in terms of functionality and ease of use. The user interface is a large part of any application or piece of software because if it is not set up correctly, the user will have a hard time setting up or using the software, rendering it ineffective. The training mode takes some of the guesswork away from Sean because it essentially tells the system which voice it should not be focused on which should allow for faster processing times. It also gives the user a sense of personalization for having their own voice profile. The environment selection section aids both the processor and microphone

array because it tells these components what to expect from the surroundings of the user. Lastly there are a number of smaller features in addition the ones discussed about that will all be discussed in the following sections. Below Figure XX shows a simple class diagram of the required functions for Sean to run effectively.

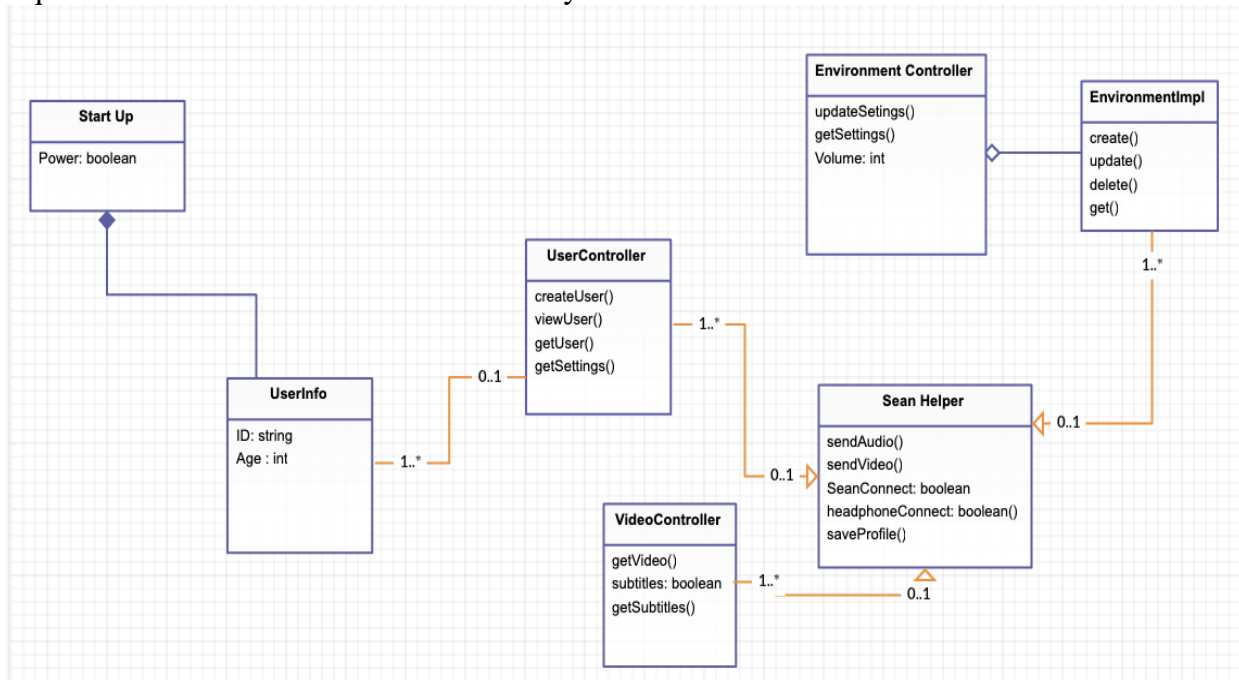


Figure 43. Simple class diagram of the mobile application.

### 6.3.1 User Interface

Upon clicking the icon for the mobile application on their device, the user will be taken to an introductory screen that displays the full name of the device as well as the logo. The logo is a graphic that depicts a simplified ear surrounded by small yellow circles, similar to what the microphone array looks like, with the name Sean to make it easily identifiable. This will give the application time to load and serve as a sort of preparation screen. Following that the user will be taken to a home screen which includes a number of elements to interact with. On the top left there will be a button that takes the user to an area to create or select an existing user profile. This is one of the first things the user should interact with because it will allow for little to no set up time for future uses of the application. To the right of that is a button that allows the user to select the environment in which he or she will be operating the device. To the right of that is the button that will the user to access Video Mode. The user does not have to use Video Mode while using Sean, and can instead remain on the home screen if he or she chooses to do so. Near the center of the screen is a sound icon with a sliding bar underneath that will control the level of audio the user experiences. However, the user could also use the buttons on their device or headphones to control the volume as well. At the bottom of the screen will be a status bar that will notify the user whether or not Sean is connected and whether or not there are headphones connected through the application. It is important that the user can see the status of both of these in the same place so they know they are using Sean properly and to allow them to troubleshoot if necessary.



Figure 44. Screen displays the introductory screen of the mobile application.

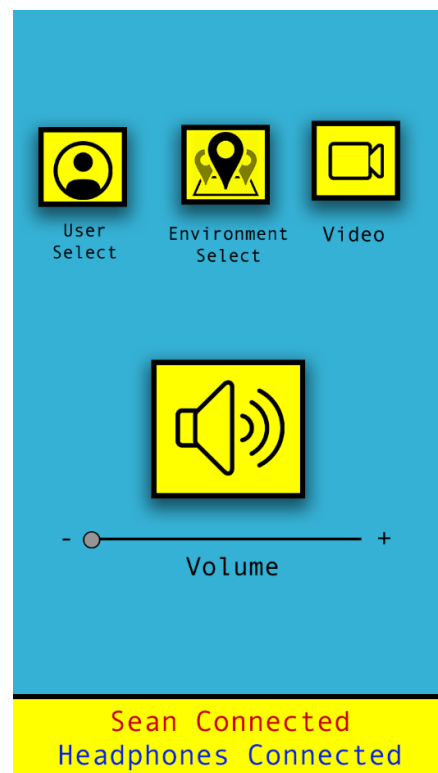
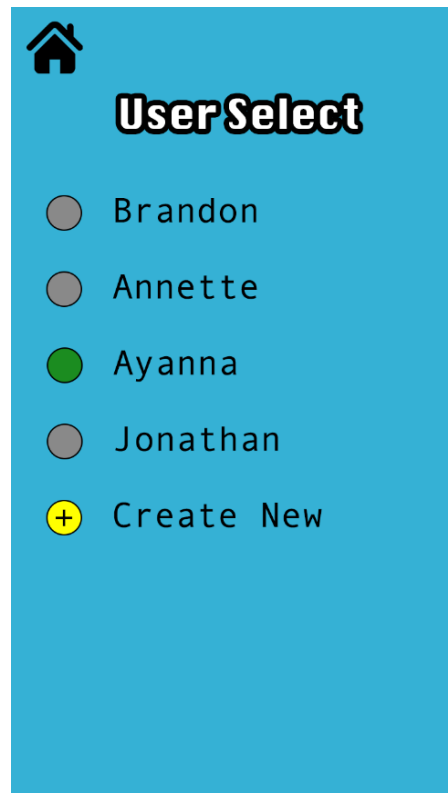
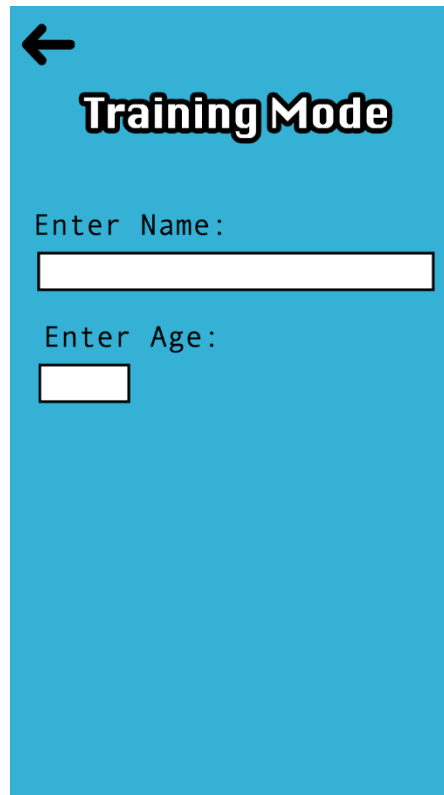


Figure 45. Screen displays the home screen of the mobile application with user select, video mode select and volume control options (right).

Once the user clicks the user select icon, they will be taken to the main user select screen. This screen lists all the user voice profiles that are stored on this device. Each profile will stay on the device until the user decides to delete it. When the user selects a profile, it will be indicated on the screen then the user may return to the home page by using the home icon in the top left corner of the screen. At the bottom there is an option to create a new voice profile. Once the user selects Create New, they will be taken into the Training Mode.



*Figure 45. Screen displays the User Select screen which allows the user to either create a new voice profile or select an existing one.*



*Figure 46. The screen above displays the first screen for training Sean on a user's voice, which prompts the user for their name and age (right).*

When the user is taken into the training mode, it will be indicated by text across the screen, so the user knows why they are being requested to do certain actions or input information. The application will prompt the user to input their name, used to identify and differentiate between user profiles, and age, to assist in deciding an appropriate range to pick up the voice of a user.

The next training screen prompts the user to say a simple phrase to configure the device to pick up the correct voice among the many that may be in one environment. They may be asked to repeat the phrase more than once to make sure Sean is identifying the voice correctly. There will be a status bar at the bottom of the screen indicating that Sean is listening to the user. Following the completion of this step, the user will be notified by another screen that the profile has been created and the name they entered will be displayed on the screen. Then the user will be taken back to the user select screen and they will be able to select the profile they just created.

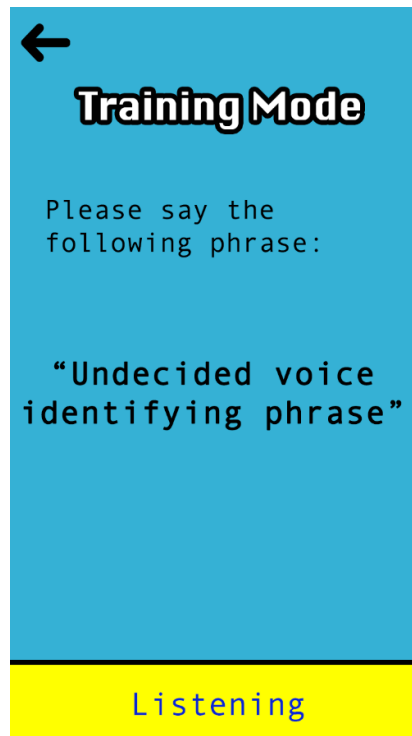


Figure 47. This screen above displays the second screen for training Sean on a user’s voice, which prompts the user to say a phrase.

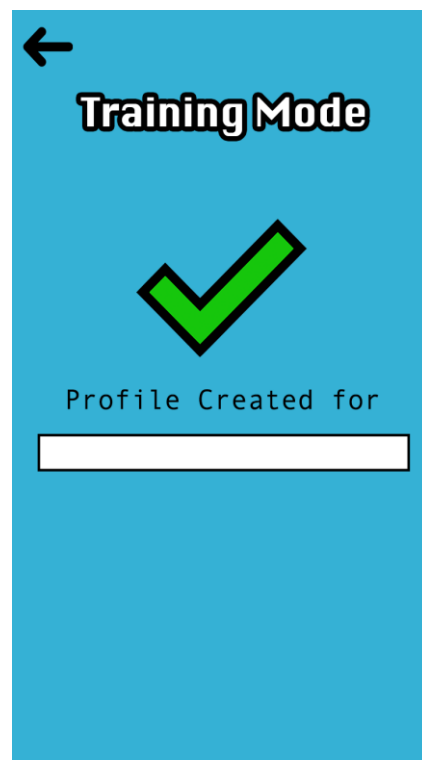
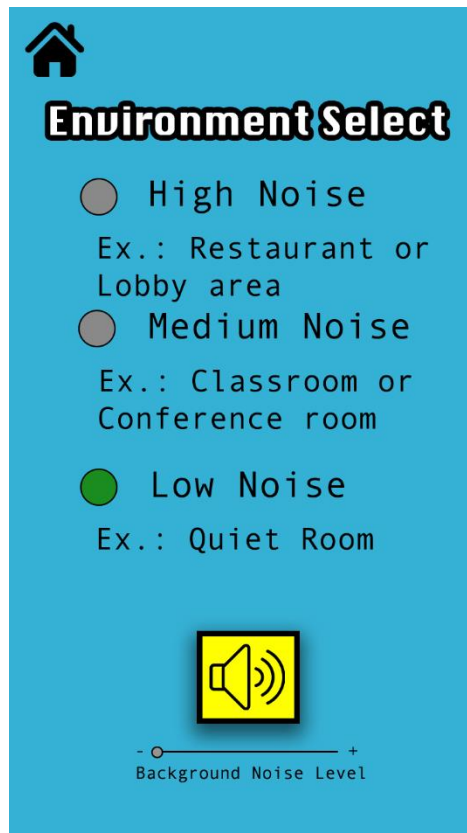


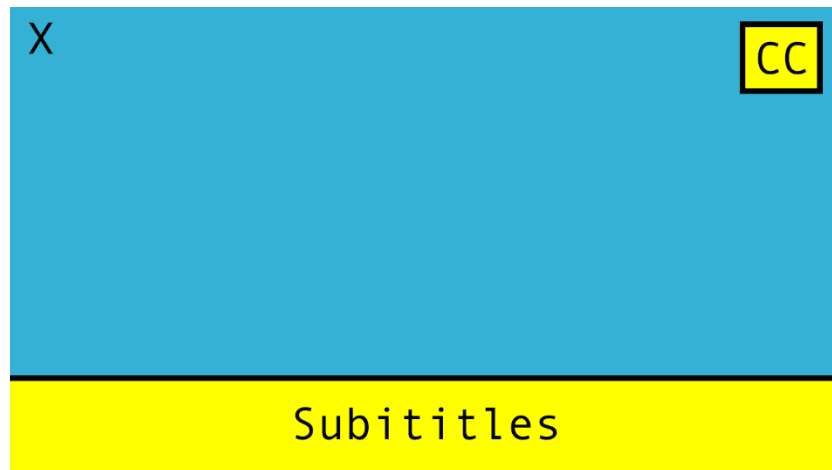
Figure 48. This screen above displays the final screen for training Sean on a user’s voice, which notifies the user that their profile has been successfully created (right).

From the home screen, the user will have the option to select the environment in which they will use Sean. There will be three preset options of a Low, Medium, or High Noise environment with examples listed to allow the user to make the appropriate selection. By default, Low Noise mode will be selected unless the user chooses otherwise. In order to provide additional customization for the user, a feature on the environment select screen is the option to choose the level of background noise the user wishes to hear. If the background noise is too distracting to a user they may set it to a minimum amount. If the user wishes for a natural audio experience, the user may choose a higher sound level.



*Figure 49. This screen above displays the Environment Select screen which allows the user to select the environment to use Sean in.*

From the home screen, the user will have the option to select video mode which changes the orientation of the application to a landscape mode. In this mode the user will be able to see the streaming video feed from the camera in Sean. There will be an option in the top corner to turn on or off subtitles for the audio that the user will hear. There will also be an “X” button will take the user back to the home screen when pressed.



*Figure 50. This screen above displays the screen for Video Mode which allows the user to stream the video from the camera in Sean and view subtitles.*

### **6.3.2 Training Mode**

In order for Sean to properly recognize a user's voice and not confuse it with the person a user may be talking to, there must be a way for the system to identify who the user is. Many applications and devices use voice biometrics for security purposes as well as user operation. Similarly, Sean will use it for identity purposes. Voice recognition software functions in two ways, either by recognizing individual words or by speech patterns in a person's voice. By prompting the user with a specific phrase, it allows the system to solely focus on the speech patterns of the user rather than guessing what the user is saying. The user will be prompted to repeat the phrase in order to create an accurate voice blueprint for the user, which will then be stored for later use.

### **6.3.3 User Environment Classification**

Various environments will require different processing specifications and different mixing requirements. High noise environments will require the processor to be more tedious in filter out noise and picking out the voice that the user wishes to hear. Whereas, Low noise environments may be as simple as just amplifying the only sound in the room, namely the other person's voice.

### **6.3.4 Other Relevant Features**

One of the main but also most basic functions of the mobile application will be creating a gateway to connect headphones to Sean. Since the application will be used on a smartphone that is Bluetooth® compatible it will allow the user to use wireless Bluetooth® headphones with the system as well. Although Bluetooth® has the ability to connect one device to multiple devices at the same time, since the video will be received, processed and sent to the application, it makes the most sense to have audio follow the same path for the purpose of synchronization. Another feature is volume control. With the use of a mobile application the user will have a number of options for volume control. The user can use the physical buttons on the device, the on screen volume control slider option or, if the headphones allow for it, the user may use the volume control that the headphones may come with. This is a real time adjustable feature that allows the user to adapt to their environment and experience for their own needs and comfort.



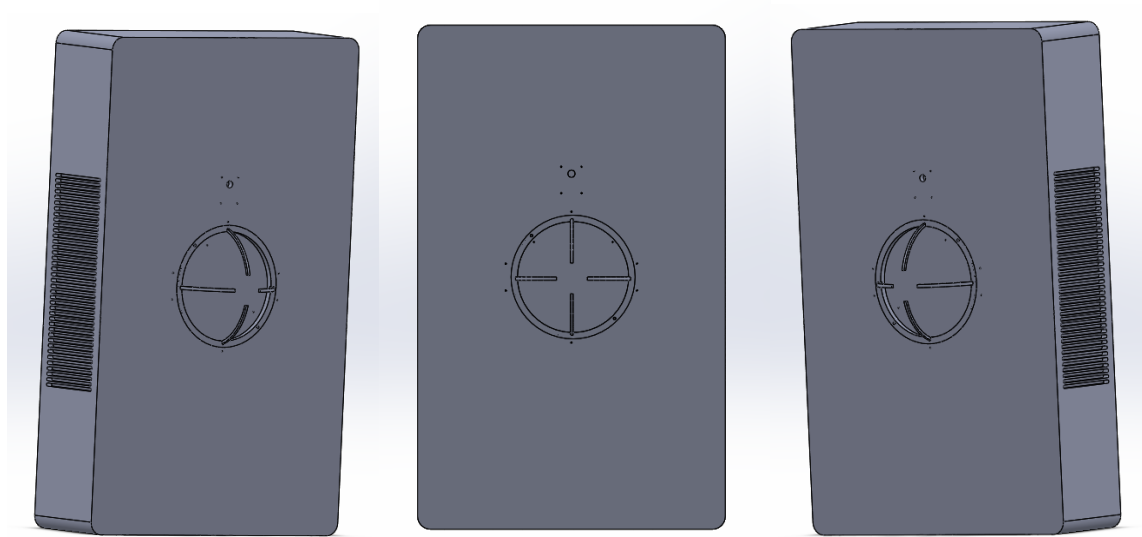
### 6.3.5 Speech to Text API

Apple has a framework built into the operating system called Speech. This was released in 2016 and it allows the user to process speech and turn it into text. This is the same process that Siri in iOS devices would use to fulfill a user's request. This API can be directly implemented through the iOS developer software XCode. Not only does the framework support speech to text, it also allows for the use of punctuation if the user so chooses. Commands include new line, new paragraph, and caps if desired. This software will be integral to the Video Mode for Sean because it has the option of displaying subtitles while streaming video from the camera embedded in Sean. Since it is an optional feature that the user may turn on or off, having the API easily accessible and compatible with the application development software will be important.

### 6.4 Packaging

To design the packaging for Sean the dimensions and masses of the components need to be known. These can be found in Table 17 below. Sean will be housed in a 3D printed enclosure made of Acrylonitrile Butadiene Styrene which is also known as ABS. It is one of the most used and longest used materials for 3D printing because its positive attributes across the board. It is low in cost which makes it ideal for small business or groups that want to use 3D printing in a project but may not have a large budget to do so. 3D printing itself is already more cost effective than most other methods of creating packaging by hand so a low cost material is an added bonus. With any device that is intended to be portable and used often, a large concern is damage. With portable devices even placing it down the wrong way could be detrimental to the machine so having a sturdy material is important. Even if the user is extremely careful and gentle with a device, using it often will create some sort of wear and tear. Therefore, ABS is a great material for Sean because it Sean is intended to be used as an everyday device that the user can carry around with them. ABS is impact resistant and not as prone to wear so it will keep the parts and machinery inside Sean safe while also keeping the outside enclosure from looking or becoming worn after only a few uses.

Table 17 -- Dimensions of Sean Components				
Component	Length (mm)	Width (mm)	Height (mm)	Mass (g)
TX2	170.18	170.18	-	56.7
Microphone Array	d=79.76	-	-	1540
Power Supply	185.42	124.46	20.32	562.5
Camera				21
Cooling Fan (2)	80	80	25	



*Figure 51. Different angles of the preliminary model and design of Sean packaging.*

In Figure 50 above, the packaging design for Sean was modeled in SOLIDWORKS®. The enclosure for Sean will be split up into three parts. The first part will be a hollowed out rectangular prism shape, with one face missing. The rectangular base of the packaging will contain the battery, the NVIDIA Jetson TX2 development kit, the PCB, and the camera. The face that is not missing will be against the user with some added padding between it and the user for comfort. The power supply and the Nvidia TX2 will be mounted to this piece using screws to hold them in place. Along the sides of this piece there will also be vents for cooling which will be discussed in a following section.

The next piece of Sean will be the other face of the rectangular prism, which will have the microphone array and camera mounted to it also using screws. In order to ensure access to the internal machinery, there will be hinges and a clasp to connect this face to the first part of Sean mentioned above. This will allow adjustments to easily be made without have to constantly screw and unscrew the top, which would eventually loosen the connection.

The circular cage on the front of the model will contain the microphone array and will have a nylon or foam cover to prevent wind or voice blasts from altering the audio. This section of the packaging will users of Sean to see the LEDs' color which is indicative of what state Sean is in (configuration, powered on,...). The largest hole above the microphone array cage is for the camera lens. All of the other holes are mounting hole which will discussed more at length later in this section. Dimensions of the packaging seen in the figure above can be found below in Table 18.

Table 18 -- Dimensions of Sean Packaging					
Component	Part Name	Length (mm)	Width (mm)	Height (mm)	Thickness (mm)
Rectangular casing	Housing for processor, PCB, and battery	394.46	240	76.20	5
Microphone cage	Windscreen cover for microphone array	d=96.78	-	17.78	2

Along with designing packaging that can properly enclose all of the hardware in this project it is imperative that these component are properly mounted within the packaging. The mounting holes are shown below in Figure 51. The yellow, blue, and orange holes are on the front of the packaging. The red and green holes are on the back of the packaging. The yellow blue, and orange correspond with the camera mount, microphone array mount, and microphone cage mount. The red and green correspond to the battery mount and the TX2 mount, respectively. Mounting holes have not yet been added for the fans that will be used for cooling Sean.

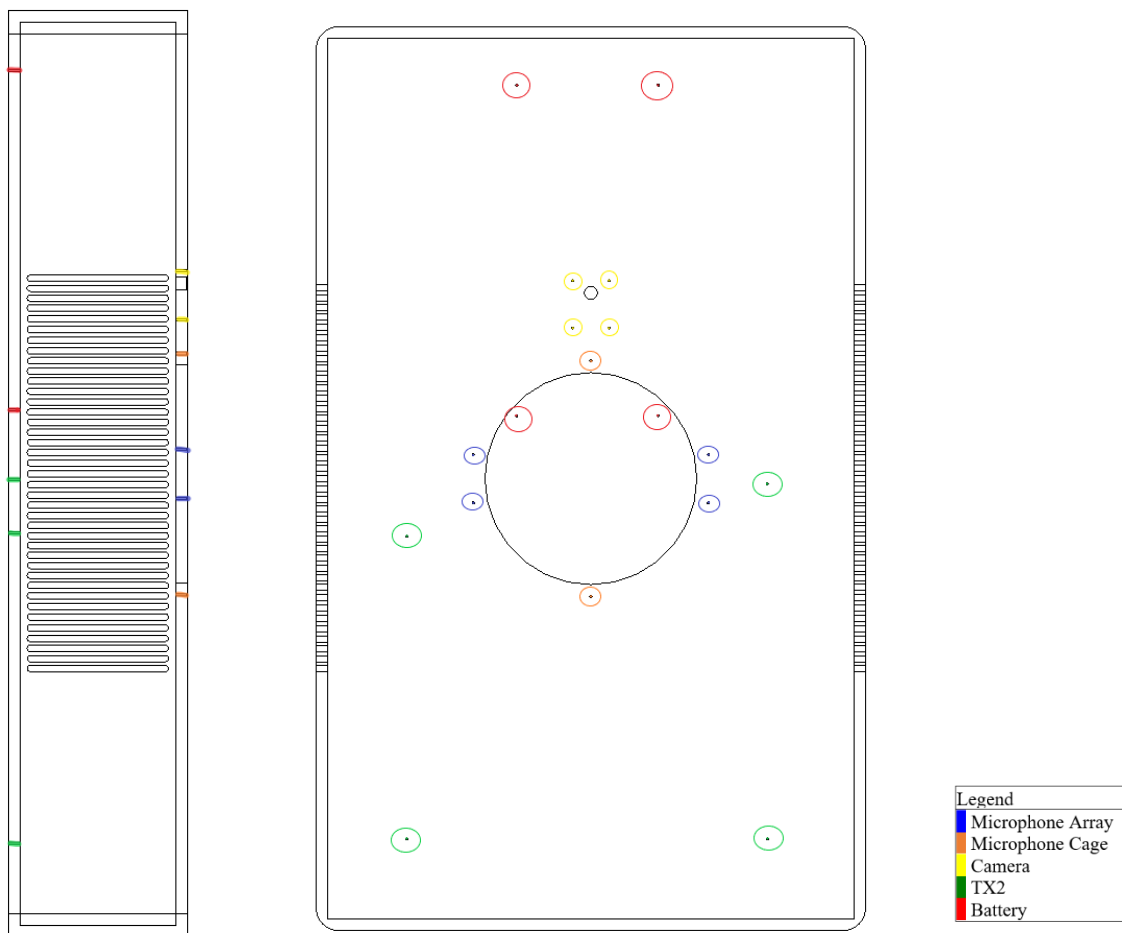
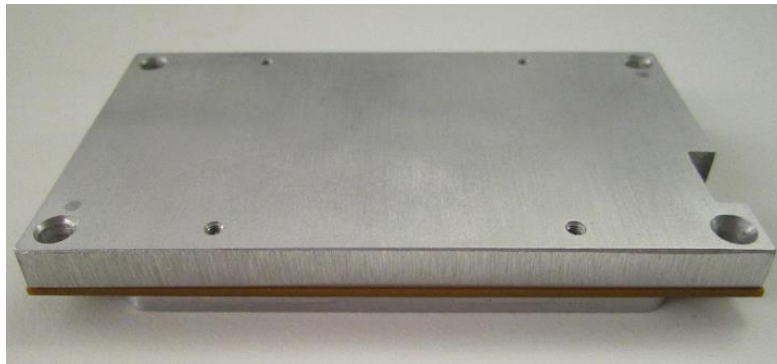


Figure 52. Mounting hole diagram for Sean packaging.

### 6.4.1 Cooling

The TX2 comes with two built in methods of cooling. Since Sean will demand a large amount of processing from the TX2 while being worn by a user, it is imperative to keep the system as cool as possible to prevent damage to the machine or injury to the user. The cooling methods used in the TX2 make up almost a fourth of the weight of the unit so it is clear that cooling the system is an important part of operation. One of the methods is the TX2 thermal transfer plate which is made out of an aluminum alloy. The maximum operating temperature of the plate is 80°C. This is an example of passive cooling. Passive cooling is known to be more energy efficient and cost effective than some other methods of temperature regulation. The other method is the fan that is attached the TX2 which pulls the hot air away from the system. This is an example of active cooling.



*Figure 53. Image above shows the passive cooling Thermal Transfer Plate that comes with the Nvidia TX2.*



*Figure 54. Image above shows the active cooling fan that comes with the Nvidia TX2.*

An additional two cooling fans will be included inside of the packaging for Sean to ensure that no unnecessary heat will linger inside of the system to cause system failure or discomfort to the user. The fans will be aligned with the slits on the sides of the packaging to allow the air to escape in the appropriate direction. They allow for USB connections so they can be directly powered by the power supply. The fans are pictured below in Figure 54.



*Figure 55. Image above shows the fans that will be used as an additional cooling mechanism for Sean.*

## **7 Integration and Testing**

All of the major components of Sean discussed in the Design section will need to be integrated into one cohesive system. The major software integration of the Audio and Visual Integrated System (AVIS) was discussed in the previous section. These will process the audio and video in parallel and communicate with each other to compare results of human sources of sound to provide an accurate analysis and execution of what human voices the user wants to hear and amplify. This kind of connection with different hardware and the computation that needs to be done in parallel will be implemented with ROS. ROS will allow for Sean to run algorithms on different hardware at the same time while providing a framework to allow for those devices to share and relay information. The output (audio and video time aligned as much as possible) is sent via Bluetooth® to the Sean app on the user's mobile phone. The mobile app will include a live video stream of what Sean can see and will send the audio via headphone jack/Bluetooth® to the user's choice of headphones.

Throughout the design and implementation of Sean, there will be two official rounds of testing: preliminary and final. Of course, testing will occur during the prototyping of Sean but the preliminary testing and final testing will allow for the group to ensure that objectives are realistic and can be met. The preliminary testing will be to ensure the implementation of our design is on track to be successful by testing the major components of the system before prototyping to confirm they meet Sean's needs and objectives. The final testing will allow for Sean's operating mode to be run through different trials to demonstrate its capabilities and results.

### **7.1 ROS and AVIS**

ROS will most likely be the fundamental control of our AVIS (Audio and Visual Integrated System). The audio and visual process will be occurring simultaneously and will need a way to communicate with each other in an efficient way. ROS provides the framework and heavy lifting to get cross-modality communication between devices that may use different programming languages for configuration. The NVIDIA Jetson TX2 and the Microphone array will be able to communicate with each other through the use of 'messages' in the ROS framework.

## **7.2 Mobile App and Device Communications**

The following sections discuss how our mobile application will communicate with AVIS and it compares whether Wi-Fi or Bluetooth® would better for streaming video and audio at the same time considering both cost (time and money) and latency.

### **7.2.1 Streaming Audio or Video: WiFi or Bluetooth®**

When people think of wireless audio the first instinct is assume that it will be streamed through Bluetooth®. Whether through a car speaker or a pair of wireless headphones, it is almost expected for it to be used with a Bluetooth® enabled device. Since Bluetooth® transmits through radio waves, the data is compressed before it is sent. Then the device sends the audio over through a narrow bandwidth to the receiving device. The problem with this is that most files are already compressed when the first Bluetooth® device receives it already so it just becomes compressed even more which ruins the quality. With WiFi, the receiving device directly accesses the audio through the internet, which means the audio does not get compressed and it allows for a higher sound quality. This is known as “lossless codec.”

When it comes to streaming video, there are advantages and disadvantages to both techniques of using Wi-Fi or Bluetooth®. One of the most obvious differences between the two is distance. Wi-Fi streaming will continue as long as you are in range of the router, which can greatly vary depending on how strong your router is. Bluetooth® will only work effectively up to about 30 feet. However, with Bluetooth®, it is a direct connection between the two devices, and Wi-Fi must include a router to establish a connection. This could pose a problem because not only would the system be relying on the cooperation between the phone and SEAN, but it would have to rely on a solid Wi-Fi connection as well. Another difference is compatibility. Bluetooth® is compatible with a vast number of devices and it can guarantee a strong connection. However not all devices are Wi-Fi enabled, and if they are the connection available may not be strong enough to properly transmit the data that is intended to be sent. The general solution to streaming problems is to go with Wi-Fi if you prefer quality or to go with Bluetooth® if you want something cheap and easy.

## **7.3 Testing**

Sean will go under testing in both Senior Design I and Senior Design II. The preliminary testing will be conducted before the beginning of Senior Design II to ensure that the major components of the system are set up and to be ready for configuration in Senior Design II--no merging of spaces yet--to allow time for us to make any needed changes early on. The second stage of testing will occur all through the first few weeks of Senior Design II to get the individual sub-systems of Sean working independently. The third stage of testing will be integration testing: the merging of systems and construction of Sean will be in progress and end by the second week of July. The final stage of testing will occur after all troubleshooting independent system and integrated system troubleshooting is complete so that the project demonstration test cases can be run through and tested for any inaccuracies. The final test of the system will be at the Project Demonstration at the end of Senior Design 2. Sean will be tested throughout the prototyping process to ensure that when we add on new components and increase the complexity of our system, we are able to quickly identify an error (software or hardware related) and spend more time on fixing the error rather than looking for it. Given the complexity of the system integration, troubleshooting will be an enormous part of prototyping.

### 7.3.1 Preliminary Testing

Major components of Sean will be tested before any software or integration testing can begin. These are baseline tests to ensure compatibility and assess limitations of all components prior to implementation. Components of the audio, visual, mobile app, and power systems will all be included in the preliminary testing. This section of testing should be complete before Senior Design II to ensure that most of time in Senior Design II will be focused on software development and troubleshooting. The action items for the preliminary testing can be found in Table 19 below.

Table 19 – Preliminary Testing	
Test	Expected Completion Date
Configure TX2	April 1
Live stream video from camera via TX2; observe results on monitor	April 10
Save video files on TX2; confirm files are saved	April 10
Connect Matrix Voice to Raspberry Pi 3 to test open-source algorithms	April 20
Test distance limitations of microphone array	April 20
Obtain and install iOS mobile application development software	May 1
Create initial mobile application that functions on an iOS device	May 1
Confirm TX2 is sufficiently powered by battery	May 1

### 7.3.2 Independent System Testing

The second stage of testing will consist of the design and troubleshooting of the individual systems that make up Sean as a whole. Sean is separated into the visual system, the audio system, and the mobile application. Each of these will have action items/successful tests that all need to be completed prior to integrated testing. The action items are based on what was detailed in section 6.2 Software Design.

Getting the visual side of the system running as expected is going to be a crucial step in the design process. Testing progress along the way is important to understand any failures quickly and correct them. For the visual system there are several test events that when completed will

show an overall progress towards Sean’s final goal. These test events are found in Table 20 below.

<b>Table 20 – Independent Visual System Testing</b>	
<b>Test</b>	<b>Expected Completion Date</b>
Complete experiments to determine best human face detection algorithm assessing the trade offs between speed and performance	May 20
Integrate live video feed with the best face detection algorithm	May 27
Identify separate human faces in an angle space	May 27
Understand if look angle can be determined	May 27
Produce final output with symbology overlaid	May 31
Optimize algorithms	June 27

The independent audio system testing lines up closely with the flow of the audio track in the software diagram from section 6.2. The first action item to be completed is the assessment of the algorithms that come on the Matrix Voice that are included in the Sound Enhancement Filter. Ideally, the SNR for this testing would be 25 dB or greater, but this will also be dependent on the set up and environment (speech level and background noise level) in which this is done. After this is done the User Voice Filter will be designed and tested thoroughly until it can remove the user’s voice present in the signal without distorting the primary signal. Once the audio sub-system can precisely remove the user’s voice from the signal for separate mixing, the audio signal will need to be split up to be able to mix the audio correctly, thus, the major sources present will need to be localized. This step in the testing will consist of confirming correct locations of major sources and separating their signals from the primary signal into different channels. The final step in the audio system testing before moving on to integration testing is ensuring the system can use it’s voice activity detection algorithm to correctly identify the source as human. All of these action items are summarized in Table 21 below.



<b>Table 21 – Independent Audio System Testing</b>	
<b>Test</b>	<b>Expected Completion Date</b>
Output clear audio and minimal noise from hardware (Sound Enhancement Filter)	May 3
Identify and remove user’s voice from raw audio (User Voice Filter)	May 10
Identify signal source location and separate signal from raw audio (Human Voice Detection-Beamforming and Signal Isolation)	May 14
Identify signal source as human/non-human (Human Voice Detection-Voice Activity Detection)	May 18

Since the mobile application will serve as the control center for Sean, a number of tests must be conducted to ensure it is functioning correctly. These tests are found in Table 21 below. It must first function on the iOS device it is installed on without any complications. Then it must be able to receive a number of inputs from the AVIS system without altering or losing any data that is transmitted. This will be tested by feeding it audio and video separately from a file or streamed from a computer. Then it will be tested by attempting to receive both at the same time. It also must be able to properly communicate with the user’s output device which would be either wired or wireless headphones. The user will be able to control the volume in a number of ways so this must be tested as well. The volume could be controlled through an onscreen slider, the buttons on a device, or the buttons on the user’s headphones.

<b>Table 22 – Independent Mobile Application System Testing</b>	
<b>Test</b>	<b>Expected Completion Date</b>
Start application on mobile device and verify it is operational	May 14
Connect headphones to application and acknowledge connection on screen	May 20
Receive audio input from computer	May 31
Receive video input from computer	May 31

Output audio to user's headphones	June 3
Control volume thorough multiple control methods	June 10

### 7.3.3 Integrated System Testing

This section of testing is conducted with the assumption that all of the preliminary and independent testing is successful. The action items and expected dates of completion are illustrated in Table 23 below. The initial packaging is one of the major components of Sean and needs an initial round of printing to ensure that there all parts of Sean have been taken into consideration in the design. Any inconsistencies found in the packaging will be immediately noted and correct in the SOLIDWORKS® design. The first step in the software integration will be to ensure that the Matrix Voice and Jetson TX2 are communicating as intended; the Matrix Voice and the TX2 will be comparing data gathered on potential sources and translating this data into a confidence value for each source that is indicative of how likely it is that a source is talking to the user. The next step is to ensure that Sean can successfully transmit the real-time processed audio and visual data to the mobile app. Once it is confirmed that communications between AVIS and the mobile app are successful, audio and video correctly streaming with expected latency, the algorithms used to mix the audio are fine-tuned through experimental testing. This first stage of fine-tuning is done with complete idealities: quiet environment, no packaging covering any hardware. The second stage of fine tuning occurs after the assembly of AVIS inside the packaging; the system is tested in the design and algorithms are tuned to further optimize the system. Both stages of fine tuning are done until there is little room left for the improvement of the operation of Sean.

Table 23 – Integrated System Testing	
Action Item	Expected Completion Date
Initial 3D printing of packaging parts	June 1
Reprint and the adjustment of packaging parts	June 8
Verify communication between audio and visual system (AVIS)	June 14
Establish real-time audio and visual streaming between AVIS and Mobile Application	June 20
Fine-tune integrated algorithms through experimental testing	June 27
Final assembly of packaging parts	July 4

Real world testing and additional fine tuning	July 10
---	---------

### 7.3.4 Final Product Testing

Senior Design II will consist of constant building, fixing, and testing of the prototype in its early stages of life to ensure it meets the minimum objectives. This testing will be conducted post-Integrated System testing to ensure as well as possible that there are no errors or surprises in the final presentations. This portion of the timeline will be fraught with testing and troubleshooting. Sean will be tested in several situations to determine the accuracy at which intended targets are amplified. These tests, detailed in Table 24 below, will also be demonstrated at the final presentation given no major objectives of Sean change from now until then.

Table 24 -- Final Product Tests	
Case	Desired Result
(1) User is alone, no other people, no unexpected noise.	“Natural” and clear environment noise.
(2) User is with one other person who is in front of the camera talking.	Target’s voice is amplified and clear. Background is attenuated but still clear and present.
(3) Case 2 with another person in the background not talking.	Target’s voice is amplified and clear. Background is attenuated but still clear and present.
(4) Case 2 with two people talking in the background.	Target’s voice is amplified and clear. Background is attenuated but still clear and present.
* (5) Case 4 with oscillating fan running nearby Sean.	Target’s voice is amplified and clear. Background is attenuated but still clear and present.

\*Case not finalized for demo

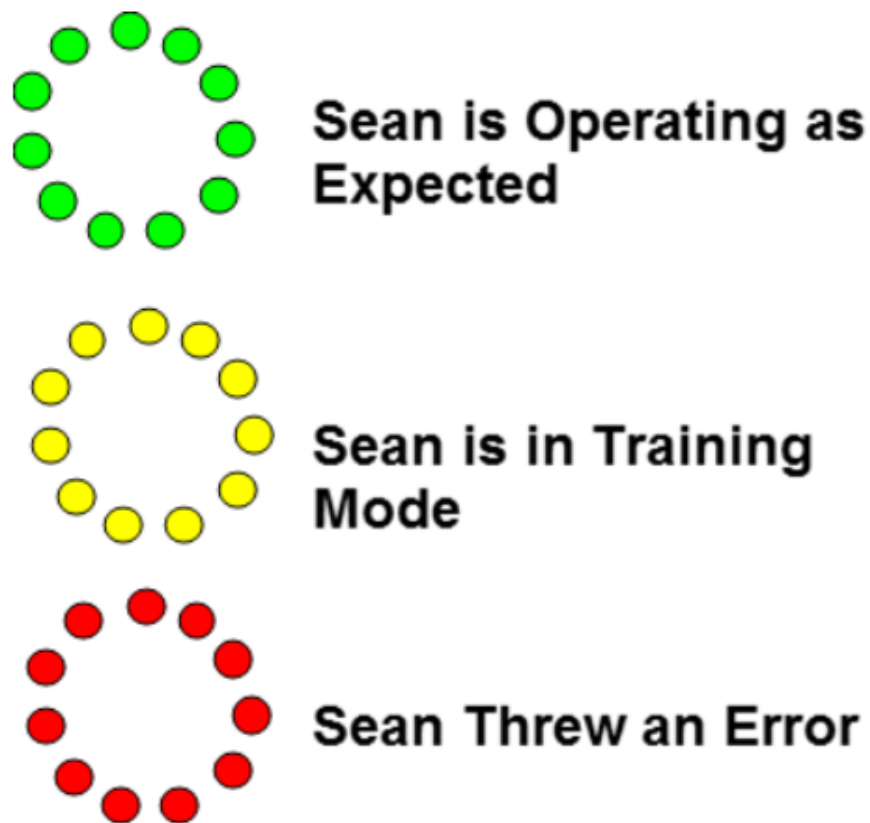
The other major tests to be conducted are with people with hearing impairments. These tests are yet to be finalized but will be so when the limitations of the system become more clear. Sean is being developed to benefit those who suffer from hearing loss; to provide any valuable data in this area of research, feedback from the users at which this is targeted to is crucial in any improvements in the design of this system.

## 8 Project Operation

In this section the training mode and normal operating modes will be discussed to allow the user to use this as a reference when setting up Sean. These will act as instruction manuals for the user and for the designers to help the user’s learn how to use the system properly to gain as much as they can out of it. This includes the technical aspects of the system to familiarize the user with

how the system works so that it can be used in an ideal way to optimize its functionalities. For instance, if the user is hearing impaired and knows that the direction they look is a factor when amplifying the desired signal source then they can take advantage of that knowledge and be sure to look as directly as they can to the source they would like amplified to. This is just one of the many ways users can provide their own idealized input into the system and achieve better results.

The last part of the section is the troubleshooting related section: the user will reference this section when Sean has identified a failure in the system or they have identified a failure in this system. This troubleshooting will be recorded in a manner that the user can potentially reference if the system fails and pinpoint where the system went wrong. Sean will also be helping the user and those configuring it to identify failures in the system by changing the color of its LEDs: Sean's LED ring will change colors based on whether it is in a normal operating mode, training mode, or has experience some failure.



*Figure 56. These LEDs indicate what mode Sean is currently operating in. Green LEDs mean sean is working as expected, Yellow LEDs indicate Sean is in training mode, and Red LEDs show that there is some sort of error.*

## 8.1 Instruction Manual

1. For first use:
2. To begin using Sean the user must first download the mobile application onto thier device. To ensure correct installation, the user will see the Sean logo after launching the application.

3. The user will be directed to the home screen. Ensure Sean is powered on and connected by observing the status bar at the bottom of the screen. If the screen displays Sean is not turned on or connected, check that Sean is powered on and verify Bluetooth is turned for the mobile device.
4. Ensure headphones are connected by observing the status bar on the bottom of the screen. If the screen displays that the headphones are not connected, check the possible issues with the headphones. If wired, the headphones may not be plugged in. If wireless, the headphones may not be powered on or Bluetooth may not be enabled on the mobile device or the headphones.
5. To configure the user profile, choose user select on the home screen. Follow instructions on the screen to begin Training Mode and enter the information prompted.
6. First enter the name and age of the user. Next, repeat the phrase displayed on the screen while observing the status bar across the bottom of the screen to ensure the device is listening.
7. When the user profile is confirmed to be created, return to the home screen. To ensure the profile just created is selected, return to the profile select screen and confirm the bubble next to the selected user is highlighted. Then return to the home screen.
8. Choose the Environment Select icon. The default environment is Low noise. Review the examples given for each choice and determine which environment is the most appropriate. Ensure the correct environment is selected by observing the change in color of the bubble next to the selected environment.
9. Observe the position of the slider on the Environment Select screen. Adjust to desire level of background noise by moving the slider to the right or left. Return to the home screen.
10. Observe the volume slider on the home screen. Adjust the volume level to desired position by sliding right and left.
11. Select Video Mode from the home screen. Turn the mobile device for proper viewing. Choose the button on the screen to enable or disable subtitles.
12. Return to the home screen by using the X button in the corner of the screen.
13. When finished, close the application and ensure Sean is powered off.

For use after initial configuration:

1. Open the Sean mobile application and observe that the logo appears on the screen.
2. Ensure Sean and headphones are connected by following steps 3-4 in the previous section.
3. Select the User Select icon from the home screen. If the desired user profile appears, ensure it is selected by the corresponding bubble being highlighted. If it is not, select Create New User and follow steps 5-7 in the previous section.
4. To select the environment, follow steps 8-9 in the previous section.
5. Observe the volume slider on the home screen. Adjust the volume level to desired position by sliding right and left.
6. Select Video Mode from the home screen. Turn the mobile device for proper viewing. Choose the button on the screen to enable or disable subtitles.
7. Return to the home screen by using the X button in the corner of the screen.
8. When finished, close the application and ensure Sean is powered off.

## 9 Administrative Content

This section contains administrative content related to the research and design of Sean. It includes the milestones which provides an overview of deadlines and expectations on progress for the project through both Senior Design I and Senior Design II. A budget analysis is included as well financing, budgeting, and any related information is included. This section includes a Project Status Diagram section that goes into detail of what major parts of the Research and Design are completed, in progress, or not begun.

### 9.1 Milestones

In Tables 25 and 26 below, the project milestones--both class and non-class related--are shown in chronological order. These provide structure to the research and design process from the beginning of Senior Design I to the end of Senior Design II. Table 25 illustrates the deadlines created for the designers for Senior Design I. These milestones were major marks in the research and design of this project are imperative to the implementation of Sean in Senior Design II

Table 25 -- Senior Design I Milestones		
Description	Duration	Dates
Divide and Conquer V1*		January 31
Divide and Conquer V2*		February 14
Decide packaging lead		February 19
Research CV, App, DSP	2 weeks	February 15-March 1
Initial algorithm trades		March 1
Come up with hard specs		March 1
Make decision about features/necessity of app		March 1
Determine PCB Function		March 1
First round parts procurement	3 weeks	March 1-March 22
Software Development	3 weeks	March 18-April 11
Finalize parts trades	2 weeks	March 22-March 31
Integration of System	2 weeks	March 22-April 11
Mid-point Draft*		March 28
Solidworks Model		April 5

Matrix and TX2 Comms		April 5
Initial TX2 Configuration		March 31
Second round parts procurement		April 11
Final Draft*		April 11
Final Document*		April 21

Table 26 illustrates the deadlines created for the designers for Senior Design I. These milestones were major marks in the research and design of this project are imperative to the implementation of Sean in Senior Design II

Table 26 -- Senior Design II Milestones		
Description	Duration	Dates
Build Prototype	7 weeks	May 13-June 24
Test, Redesign, Test	3 weeks	
Contact SD panel	1 week	
Final Prototype	2 weeks	June 24-July 8
Peer Presentation*		TBA
Final Report*		TBA
Final Presentation*		TBA

\*Class-related milestones

## 9.2 Budget Analysis

This project is being sponsored by Lockheed Martin. Table 27 shown below is a revised budget of the parts that will be bought to complete the project from a budget of \$2000 allocated to the group from the sponsorship. The NVIDIA Jetson TX2 has been provided to the group for use in the project by Lockheed Martin and is thus exempt from the budget. The following prices reflect each part but they do not include any possible necessities for reorder due to testing to ineffectiveness of parts, but these issues have also been considered when developing our budget.

Table 27 -- Actual Budget				
Item	Part Name	Price estimate	Lead Time	Obtained?
Microphone array	<a href="#">Matrix Voice</a>	\$61.05	2 business days	✓
Camera	<a href="#">e-CAM132 TX2 - 13 MP MIPI Autofocus Camera</a>	\$284.00	4 weeks	
Tripod	Amazon	\$20.00		
PCB**	Advanced Circuits/Silver Circuits	\$30.00		
Packaging	Customized based on technology needs	\$150.00		
Wearable Item	<a href="#">GoPro Chest Strap</a>	\$8.99	2 days	
Power Source	<a href="#">POWERADD Pilot Pro2 23000mAh Power Bank 4.5A DC</a>	\$80.00	2 days	✓
Processors	Nvidia Jetson TX2	\$0.00 (Provided by Lockheed Martin)		✓
USB Cooling Fans	Coolerguys Dual 80mm USB Cooling Fans	\$13.95		
Apple Developer Program Fee	-	\$105.93		✓
Microphone Array Testing Device	Raspberry Pi 3	\$37.27		✓
Device Memory	16 GB Sandisk Micro SD Card	\$10.54		✓
Thermal Sensor Circuit	PCB	\$30.00		
Thermal Sensor	LM 35 Thermal Sensor	\$0.84		
<b>Total:</b>	\$2,000	\$801.73		

### 9.3 Project Status Diagram

As progress is made throughout the project the below diagram outlining team member responsibilities relating to the overall system are shown below. How the status of these high-level blocks are updated is a function of overall progress on that section. Currently, all research has been completed on the audio, visual, and power sides of the system. Brandon Kessler and



Annette Barboza are currently working through finalizing what algorithms will be used in the software design and how they will be implemented. Annette and Brandon will be working together to merge the audio and visual spaces of AVIS and then use deep-learning practices to develop a system that can accurately predict what person the user wants to listen to. Ayanna Ivey is working on AVIS communications to the Sean mobile application while also designing it completely to meet the needs of Sean and provide a user friendly and intuitive interface for users of this design. This is illustrated in Figure 56 below.

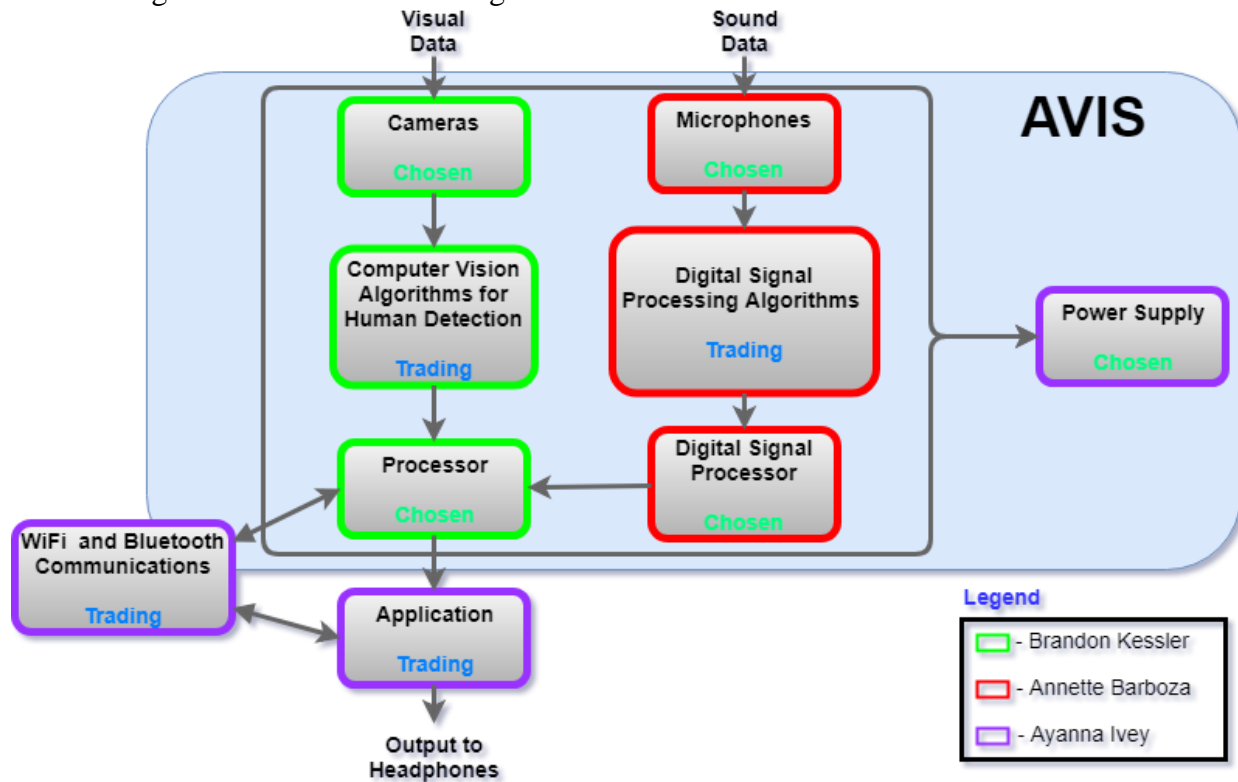


Figure 57. Project Status diagram relating to the overall structure of the system with team member focused areas shown.

## Appendix A References

1. Quick Statistics About Hearing. (2018, October 05). Retrieved from <https://www.nidcd.nih.gov/health/statistics/quick-statistics-hearing>
2. Hearing aids: How to choose the right one. (2018, May 22). Retrieved from <https://www.mayoclinic.org/diseases-conditions/hearing-loss/in-depth/hearing-aids/art-20044116>
3. [http://www.ecs.umass.edu/ece/sdp/sdp17/team10/app/res/FPR\\_report.pdf](http://www.ecs.umass.edu/ece/sdp/sdp17/team10/app/res/FPR_report.pdf)
4. Hearing aids: How to choose the right one. (2018, May 22). Retrieved from <https://www.mayoclinic.org/diseases-conditions/hearing-loss/in-depth/hearing-aids/art-20044116>
5. <https://www.invensense.com/wp-content/uploads/2015/02/AN-1112-v1.1.pdf>
6. <https://www.digikey.com/product-detail/en/knowles/SPH1668LM4H-1/423-1404-1-ND/5332433>
7. Heurig R, Chalupper J. Acceptable Processing Delay in Digital Hearing Aids. *Hearing Review*. 2010;17(1):28-31.
8. [https://www.researchgate.net/publication/261211441\\_A\\_lip\\_reading\\_application\\_on\\_MS\\_Kinect\\_camera](https://www.researchgate.net/publication/261211441_A_lip_reading_application_on_MS_Kinect_camera)
9. <http://www.hearingreview.com/2013/03/designing-hearing-aid-technology-to-support-benefits-in-demanding-situations-part-1/>
10. <https://www.hearingtracker.com/best-hearing-aid-brands-in-2019>
11. <http://www.bmva.org/visionoverview>
12. Vidanapathirana, M., & Vidanapathirana, M. (2018, March 24). Real-time Human Detection in Computer Vision - Part 1. Retrieved from <https://medium.com/@madhawavidanapathirana/https-medium-com-madhawavidanapathirana-real-time-human-detection-in-computer-vision-part-1-2acb851f4e55>
13. Shaikh, F., & Faizan. (2018, July 13). Understanding and Building an Object Detection Model from Scratch in Python. Retrieved from <https://www.analyticsvidhya.com/blog/2018/06/understanding-building-object-detection-model-python/>
14. Human face detection in a complex background. (2003, May 19). Retrieved from <https://www.sciencedirect.com/science/article/pii/S0031320394900175?via=ihub>
15. [https://www.researchgate.net/publication/257338580\\_A\\_Review\\_on\\_Face\\_Detection\\_Methods](https://www.researchgate.net/publication/257338580_A_Review_on_Face_Detection_Methods)
16. Deep Learning Haar Cascade Explained. (2018, August 17). Retrieved from <http://www.willberger.org/cascade-haar-explained/>
17. <https://www.cs.cmu.edu/~efros/courses/LBMV07/Papers/viola-cvpr-01.pdf>
18. Olena, & Olena. (2018, February 08). GPU vs CPU Computing: What to choose? Retrieved from <https://medium.com/altumea/gpu-vs-cpu-computing-what-to-choose-a9788a2370c4>
19. What's the Difference Between a CPU and a GPU? (2019, March 07). Retrieved from <https://blogs.nvidia.com/blog/2009/12/16/whats-the-difference-between-a-cpu-and-a-gpu/>
20. <https://developer.nvidia.com/embedded/buy/jetson-agx-xavier-devkit>

21. [https://developer.ridgerun.com/wiki/index.php?title=Xavier/Processors/HDAV\\_Subsystem/Audio\\_Engine](https://developer.ridgerun.com/wiki/index.php?title=Xavier/Processors/HDAV_Subsystem/Audio_Engine)
22. <https://developer.nvidia.com/embedded/buy/jetson-tx2-devkit>
23. [https://developer.ridgerun.com/wiki/index.php?title=ASoC\\_Driver\\_in\\_Jeston\\_TX1\\_and\\_TX2#Jetson\\_TX1.2FTX2\\_audio\\_subsystem\\_7C\\_based\\_in\\_TRM\\_chapter\\_23](https://developer.ridgerun.com/wiki/index.php?title=ASoC_Driver_in_Jeston_TX1_and_TX2#Jetson_TX1.2FTX2_audio_subsystem_7C_based_in_TRM_chapter_23)
24. <https://www.nvidia.com/en-us/autonomous-machines/embedded-systems-dev-kits-modules/>
25. <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>
26. <https://www.globenewswire.com/news-release/2018/11/26/1656453/0/en/Global-Hearing->
27. Hearing Aid Buying Guide. (2019, January 2). Retrieved March 28, 2019, from <https://www.consumerreports.org/cro/hearing-aids/buying-guide/index.htm>
28. Gupta, N., Lucia, A., Dunn, N., & Puzella, M. (2017). Earbeamer: A Parallel Beamforming Hearing Aid System. Retrieved March 28, 2019, from [http://www.ecs.umass.edu/ece/sdp/sdp17/team10/app/res/FPR\\_report.pdf](http://www.ecs.umass.edu/ece/sdp/sdp17/team10/app/res/FPR_report.pdf)
29. Gandel, C. (2016, October 03). Hearing Aid Price, How To Keep Costs Down. Retrieved March 28, 2019, from <https://www.aarp.org/health/conditions-treatments/info-2016/hearing-aid-costs-prices-cs.html>
30. 13/02/2019, A. (2019, February 15). Why Hearing Aid Prices are Inflated. Retrieved March 28, 2019, from <https://www.audicus.com/hearing-aid-price-bubble/>
31. <http://www.hearingreview.com/2018/05/real-life-applications-machine-learning-hearing-aids-2/>
32. Designing Hearing Aid Technology to Support Benefits in Demanding Situations, Part 1. (2013, March 6). Retrieved March 28, 2019, from <http://www.hearingreview.com/2013/03/designing-hearing-aid-technology-to-support-benefits-in-demanding-situations-part-1/>
33. Acoustic BF univ of windsor pdf
34. [https://www.spsc.tugraz.at/sites/default/files/Clenet10\\_DA.pdf](https://www.spsc.tugraz.at/sites/default/files/Clenet10_DA.pdf)
35. DSP delay and sum beam forming microphone array 3D sensitivity patterns. (n.d.). Retrieved March 28, 2019, from
36. Source Localization Using Generalized Cross Correlation. (n.d.). Retrieved March 28, 2019, from <https://www.mathworks.com/help/phased/examples/source-localization-using-generalized-cross-correlation.html>
37. [https://www.uio.no/studier/emner/matnat/ifi/INF5410/v12/undervisningsmateriale/foils/conventional\\_beamforming\\_part\\_1\\_plus\\_2.pdf](https://www.uio.no/studier/emner/matnat/ifi/INF5410/v12/undervisningsmateriale/foils/conventional_beamforming_part_1_plus_2.pdf)
38. <https://www.matrix.one/products/voice>
39. <https://github.com/matrix-io/matrix-creator-hal/>
40. Robert, E. (n.d.). Piconets in Bluetooth Technology. Retrieved from <https://www.electronic diary.com/2015/10/piconet-in-bluetooth-technology.html>
41. Kayne, R., & Wynn, L. S. (2019, March 10). What is Bluetooth? Retrieved from <https://www.wisegeek.com/what-is-bluetooth.htm>
42. <http://www.cs.miami.edu/home/burt/learning/Csc524.052/notes/wifi.html>
43. What is 3D printing? The definitive guide. (n.d.). Retrieved from <https://www.3dhubs.com/guides/3d-printing/>
44. Bluetooth vs WiFi Audio Streaming. (n.d.). Retrieved from <https://kmakits.com/blogs/speaker-building-how-to/bluetooth-vs-wifi-audio-streaming>

45. Entertainment, F. (n.d.). Wi-Fi vs Bluetooth: Is There an Impact on Sound Quality? Retrieved from <https://www.fusionentertainment.com/pulse/wi-fi-vs-bluetooth-is-there-an-impact-on-sound-quality>
46. Digital, M. (2016, March 08). A History of Mobile Application Development. Retrieved from <https://manifesto.co.uk/history-mobile-application-development/>
47. Lamkin, P. (2016, February 17). Wearable Tech Market To Be Worth \$34 Billion By 2020. Retrieved from <https://www.forbes.com/sites/paullamkin/2016/02/17/wearable-tech-market-to-be-worth-34-billion-by-2020/#174bb3b3cb55>
48. By. (2019, March 26). Wearable Camera Market Size 2019, Global Trends, Industry Share, Growth Drivers, Business Opportunities and Demand Forecast to 2025. Retrieved from <https://www.marketwatch.com/press-release/wearable-camera-market-size-2019-global-trends-industry-share-growth-drivers-business-opportunities-and-demand-forecast-to-2025-2019-03-26>
49. Newlands, M. (2017, February 06). The Future Of The Camera Is In Wearables: Here's Why. Retrieved from <https://www.forbes.com/sites/mnewlands/2017/02/06/the-future-of-the-camera-is-in-wearables-heres-why/#6b14eb4062fb>
50. Here Active Listening - Change The Way You Hear The World. (n.d.). Retrieved from <https://www.kickstarter.com/projects/dopplerlabs/here-active-listening-change-the-way-you-hear-the>
51. Elgan, M. (2015, August 17). New earbuds give you super-hearing. Retrieved from <https://www.computerworld.com/article/2971267/new-earbuds-give-you-super-hearing.html>
52. Lintz, J., Bill, Lintz, J., Jonathan, Gatto, Holderbaum, F., . . . Everyday Hearing. (2019, February 22). The Complete Guide to Hearable Technology in 2019. Retrieved from <https://www.everydayhearing.com/hearing-technology/articles/hearables/>
53. Gandhi, R., & Gandhi, R. (2018, July 09). R-CNN, Fast R-CNN, Faster R-CNN, YOLO - Object Detection Algorithms. Retrieved from <https://towardsdatascience.com/r-cnn-fast-r-cnn-faster-r-cnn-yolo-object-detection-algorithms-36d53571365e>
54. Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. 2014 IEEE Conference on Computer Vision and Pattern Recognition. doi:10.1109/cvpr.2014.81
55. Girshick, R. (2015). Fast R-CNN. 2015 IEEE International Conference on Computer Vision (ICCV). doi:10.1109/iccv.2015.169
56. Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6), 1137-1149. doi:10.1109/tpami.2016.2577031
57. Ademovic, A. (2016, March 03). An Introduction to Robot Operating System: The Ultimate Robot Application Framework. Retrieved from <https://www.toptal.com/robotics/introduction-to-robot-operating-system>

58. Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. doi:10.1109/cvpr.2016.91
59. Hui, J., & Hui, J. (2018, March 18). Real-time Object Detection with YOLO, YOLOv2 and now YOLOv3. Retrieved from [https://medium.com/@jonathan\\_hui/real-time-object-detection-with-yolo-yolov2-28b1b93e2088](https://medium.com/@jonathan_hui/real-time-object-detection-with-yolo-yolov2-28b1b93e2088)
60. What is Real Time Streaming Protocol (RTSP)? - Definition from WhatIs.com. (n.d.). Retrieved from <https://searchvirtualdesktop.techtarget.com/definition/Real-Time-Streaming-Protocol-RTSP>
61. Syme, M., & Goldie, P. (2004, March 5). Informit. Retrieved from <http://www.informit.com/articles/article.aspx?p=169578&seqNum=3>
62. Agarwal, T., Agarwal, T. A., Dharani, Agarwal, T., & Edgefx Technologies Pvt Ltd. (2017, March 22). Different Types of Bluetooth Technology, Working, and Its applications. Retrieved from <https://www.efxkits.us/different-types-bluetooth-technology-working-applications/>
63. The Impact of Standards on Business and Industry. (n.d.). Retrieved from <https://www.standardslearn.org/lessons.aspx?key=42>
64. <https://www.e-consystems.com/13mp-autofocus-nvidia-jetson-tx2-camera-board.asp#key-features>
65. <https://developer.nvidia.com/embedded/dlc/l4t-27-1-jetson-tx2-user-guide>
66. <https://www.mouser.com/ds/2/389/mp34db02-955149.pdf>
67. <https://circuiteasy.com/temperature-sensor/>
68. <http://www.ti.com/lit/ds/symlink/lm35.pdf>
69. <https://circuitdigest.com/electronic-circuits/temperature-controlled-leds-using-lm35>
70. <https://blog.paperspace.com/how-to-implement-a-yolo-object-detector-in-pytorch/>
71. <http://cs231n.github.io/neural-networks-1/>
72. Marr, B. (2018, December 12). What Is Deep Learning AI? A Simple Guide With 8 Practical Examples. Retrieved from <https://www.forbes.com/sites/bernardmarr/2018/10/01/what-is-deep-learning-ai-a-simple-guide-with-8-practical-examples/#4c3244c8d4ba>
73. A Beginner's Guide to Neural Networks and Deep Learning. (n.d.). Retrieved from <https://skymind.ai/wiki/neural-network><https://ujjwalkarn.me/2016/08/09/quick-intro-neural-networks/>
74. 10155233989138523. (2018, November 26). Softmax Function, Simplified. Retrieved from <https://towardsdatascience.com/softmax-function-simplified-714068bf8156><https://mattmazur.com/2015/03/17/a-step-by-step-backpropagation-example/>
75. <https://towardsdatascience.com/a-tour-of-the-top-10-algorithms-for-machine-learning-newbies-dde4edffae11>
76. Ujjwalkarn. (2017, May 29). An Intuitive Explanation of Convolutional Neural Networks. Retrieved from <https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/>
77. [https://www.researchgate.net/figure/An-example-of-convolution-operation-in-2D-2\\_fig3\\_324165524](https://www.researchgate.net/figure/An-example-of-convolution-operation-in-2D-2_fig3_324165524)
78. [https://en.wikipedia.org/wiki/Body\\_proportions](https://en.wikipedia.org/wiki/Body_proportions)

## Appendix B Permissions

All Figures are currently pending permissions and will be removed before the final version of this document if there is no permission granted.

Re: Permission to Use Figure from "A Step by Step Backpropagation Example"

Matt Mazur <matthew.h.mazur@gmail.com>

Wed 4/17/2019 10:34 PM

To: Brandon Kessler <brandon.kessler44@knights.ucf.edu>

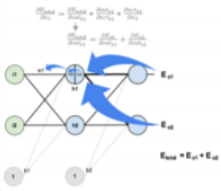
Yep go for it!

I'm in Lake smart by the way - small world!

On Wed, Apr 17 2019 at 10:07 PM, <brandon.kessler44@knights.ucf.edu> wrote:

Hello Matt,

I am writing a paper at the University of Central Florida that has a section describing deep neural networks and backpropagation. I would like to use the following figure from your article to provide the reader with an illustration of backpropagation.



Can I have permission to use this figure? Of course I will be citing you and your article.

Thank you,

Brandon Kessler

Re: KarolMajek.pl: Permission to Use Your Object Detection Picture for School Paper?

Brandon Kessler

Thu 4/18/2019 3:41 PM

Sent Items

To: Karol Majek <karolmajek@gmail.com>

Karol,

Thank you!

Brandon Kessler

From: Karol Majek <karolmajek@gmail.com>

Sent: Wednesday, April 17, 2019 11:48:13 PM

To: Brandon Kessler

Subject: Re: KarolMajek.pl: Permission to Use Your Object Detection Picture for School Paper?

Hello Brandon,

Thank you for asking!

Yes you can use it, no problem.

Check this list of state of the art: [bit.ly/object-detection](http://bit.ly/object-detection)

Regards,  
Karol Majek

On Thu, Apr 18, 2019, 02:41 Brandon Kessler <wordpress@karolmajek.pl> wrote:

From: Brandon Kessler <brandon.kessler44@knights.ucf.edu>

Subject: Permission to Use Your Object Detection Picture for School Paper?

-----  
Hello Karol,

I am writing a paper for the University of Central Florida that includes a section on modern object detection approaches using deep learning convolution neural networks. I would like to use one of your figures to illustrate object detection. It is the same one in this link: <https://blog.paperspace.com/how-to-implement-a-yolo-object-detector-in-cvtorch/>

Can I have permission to use this figure, giving you credit in my paper of course?

Thank you,

Brandon Kessler

ATS.: Permission to Use Figure from "SURVEY OF FACE DETECTION AND RECOGNITION METHODS "

Vidas Raudonis <vidas.raudonis@ktu.lt>

Thu 4/18/2019 1:19 AM

To: Brandon Kessler <brandon.kessler44@Knights.ucf.edu>;

Hi,

Ok.

Best regards

Vidas

---

**Nuo:** Brandon Kessler <brandon.kessler44@Knights.ucf.edu>

**Išsiųsta:** 2019 m. balandžio 18 d. 04:25:51

**Ik:** Raudonis Vidas

**Tema:** Permission to Use Figure from "SURVEY OF FACE DETECTION AND RECOGNITION METHODS "

Hello Professor Raudonis,

I am writing a paper at the University of Central Florida that has a section surveying human face detection techniques. I would like to use the following figure from your paper to illustrate a feature-based approach with Haar-like features.



Can I have permission to use this figure? Of course I will be citing you and your paper.

Thank you,

Brandon Kessler

Re: Permission to Use Figure from "A Tour of The Top 10 Algorithms for Machine Learning Newbies"

James Le <jl1165@rit.edu>

Thu 4/18/2019 2:11 PM

To: Brandon Kessler <brandon.kessler44@Knights.ucf.edu>;

Thanks for reaching out Brandon,

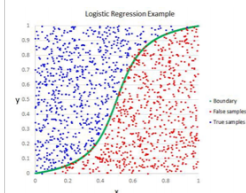
Feel free to use the image from that article.

Best,

On Wed, Apr 17, 2019 at 10:22 PM Brandon Kessler <brandon.kessler44@knights.ucf.edu> wrote:

Hello James,

I am writing a paper at the University of Central Florida that has a section describing deep neural networks and logistic regression. I would like to use the following figure from your article to provide the reader with an illustration of logistic regression.



Can I have permission to use this figure? Of course I will be citing you and your article.

Thank you,

Brandon Kessler



Anh Vo <anh.vnn810@gmail.com>  
Thu 4/18, 9:45 PM  
Brandon Kessler ✉

Hi Brandon,

Yes you can definitely use my the images as you want :D I'm glad that my post did help you.

Good luck on the paper.

--

Best regards,

Võ Nguyễn Nhật Anh (Mr.)

Re: Permission to Use YOLO figures?

Joseph Redmon <pjreddie@cs.washington.edu>

Fri 4/19/2019 2:10 PM

To: Brandon Kessler <brandon.kessler44@knights.ucf.edu>

Yes that's fine

On Apr 19, 2019, at 10:31 AM, Brandon Kessler <[brandon.kessler44@knights.ucf.edu](mailto:brandon.kessler44@knights.ucf.edu)> wrote:

Hello Joseph,

I am writing a paper at the University of Central Florida that has a section surveying detection methods. I would like to use your figures showing the architecture diagram and function of YOLO from your papers to illustrate to the reader how these architectures function.

<pastedImage.png>

<pastedImage.png>

Can I have permission to use these figures? Of course I will be citing you and your work.

Thank you,

Brandon Kessler



Re: Permission to use figure from article: "What's the Difference Between a CPU and a GPU?"

Kevin Krewell <kevin@tiriasresearch.com>

Fri 4/19/2019 3:11 PM

To: Brandon Kessler <brandon.kessler44@knights.ucf.edu>

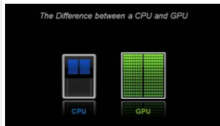
You have my permission.

Kevin Krewell

On Fri, Apr 19, 2019 at 11:43 AM Brandon Kessler <brandon.kessler44@knights.ucf.edu> wrote:

Hello Kevin,

I am writing a paper at the University of Central Florida that has a section discussing the difference between CPU and GPU computing. I would like to use your figure from your article: "What's the Difference Between a CPU and a GPU?" to give the reader a visual representation.



Can I have permission to use this figure? Of course I will be citing you and your work.

Thank you,

Brandon Kessler

Re: Permission to Use Figures from R-CNN Papers?

Ross Girshick <rbg@eecs.berkeley.edu>

Sat 4/20/2019 9:59 AM

To: Brandon Kessler <brandon.kessler44@knights.ucf.edu>

Hi Brandon,

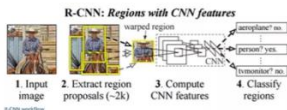
Permission is granted.

Best regards,  
Ross

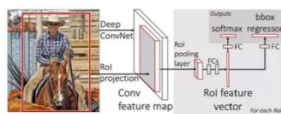
On Thu, Apr 18, 2019 at 7:30 PM Brandon Kessler <brandon.kessler44@knights.ucf.edu> wrote:

Hello Dr. Girshick,

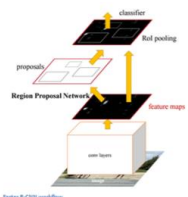
I am writing a paper at the University of Central Florida that has a section surveying detection methods. I would like to use your figures showing the architecture diagrams of R-CNN, Fast R-CNN, and Faster R-CNN from your papers to illustrate to the reader how these architectures function



R-CNN work-flow



Fast R-CNN work-flow



Faster R-CNN work-flow

Can I have permission to use these figures? Of course I will be citing you and your work.

Thank you,

Brandon Kessler