DATE: December 4, 2020

TO: Ananth Dodabalapur
Hanan Hashem
Venkata Suresh Rayudu

FROM: Evan Canter
Tanjeem Mazid
Andrea Nguyen
Deepanshi Sharma
Alexander Voth

SUBJECT: Enhanced Augmented Reality System Design Report

1.0 INTRODUCTION

This report captures team's augmented reality design and solution to providing greater accessibility. Specifically, our design seeks to address the issue of reduced sound awareness, which is a major obstacle to communication for deaf or hard of hearing individuals [1]. The final design will consist of a 3D printed head-mounted device housing an optical system, microphones, and cameras. The software of the system will be developed on a laptop computer and interfaced with the wearable device via a DSI HSMI adapter. The software will perform object detection, audio processing, captioning, and sentiment analysis to create corresponding graphics that will overlay onto the real world. In developing this design, the team identified several risks. These risks include incompatible micro display selection, obstructing the user's view, and potential inaccuracies in audio data. Our team mitigated these risks through four experiments and concluded to utilize an LCD with a least 800 nits, OpenCV's computer vision library, and Google Cloud's Speech to Text API. Additionally, the team created a project management plan for the upcoming semester. This plan includes the division of tasks from finalizing our designs to hardware assembly and software module completion.

2.0 DESIGN PROJECT BACKGROUND

Reduced sound awareness has wide-ranging impacts for deaf or hard of hearing (DHH) individuals, from missing critical notifications such as a ringing fire alarm to inconveniences like a phone ringing at an inappropriate time [1]. We seek to overcome these obstacles to communication with enhanced augmented reality technology. The purpose of our project is to merge computer data and the real world, resulting in a mixed-reality environment that will

provide hearing impaired users with a graphical visualization of sound data. This will be accomplished by harnessing optical hardware to combine computer-generated images displaying sound intensities, transcriptions, and sentiment state with what the user is experiencing in real life. The product will be a headset-mounted display consisting of an optical system and microphones that will overlay these graphical visualizations of sound onto the user's view of their environment in real time. The combination of audio data graphics and the user's perception of their real environment will constitute an augmented reality that enhances the user's ability to interact with the world around them.

2.1 Design Problem Description

While hearing aids and implanted devices can improve sound and speech recognition, they do not eliminate hearing impairment. Residual issues can include speech intelligibility, ability to interpret sound intensity, and sensitivity to background noise [5]. Furthermore, DHH individuals can also have trouble recognizing the emotional state of the speaker, which is a major obstacle to communication [14]. Our work addresses these residual issues by building on research in the sound awareness needs of DHH individuals and existing visual approaches to assistive technologies. The technology is designed to address sound urgency, identification, and sentiment analysis to provide the user with useful information about his environment.

2.1.1 Sound Awareness Needs and Communication Strategies

Understanding the sound awareness needs of DHH individuals is critical to the development of our technology. People with audial impairments use strategies such as two-way notetaking, gestures, and speechreading (which relies on visual signals such as body language, facial expressions, and lip movement to interpret speech) to communicate with others in spoken language [2]. Sound awareness needs range from identifying who or what is currently speaking/sounding to desires for real-time transcriptions. An early study led by researchers at the University of California, Berkeley and Carnegie Mellon University conducted interviews with 18 DHH participants to assess their sound awareness needs across three different contexts: at home, while at work, and in transit [2]. Across the three contexts, safety-related sounds, such as alarms and honking, were generally prioritized in interest. A more recent survey with 87 DHH participants conducted at the University of Washington confirmed these results, reporting less

concern with knowing about a phone call or alarm clock going off and greater concern for matters relating to identity and urgency, like smoke detectors or baby crying sounds [3]. Among younger DHH individuals, sound awareness needs were more related to social interactions. Researchers at Xidian University and Princeton University surveyed 60 DHH youth and adults aged 10-26 on their sound awareness needs, and they found that the primary sounds of interest were related to conversations with others [4]. The results of this study point towards a desire for captioning and emotional contextualization among younger DHH individuals. Overall, sound awareness needs vary according to context and background, ranging from knowing sound identity and urgency to desires for real-time captioning and sentiment analysis.

2.1.2 Existing Visual Sound Awareness Approaches

Many visual approaches to sound awareness focus on non-spoken language in the form of spectrographs displaying sound pitch and amplitude over time [1, 5, 6]. Graphical depictions of sound amplitude over time employ waveforms as visualizations, and they are targeted at expert users who can deduce information about environmental sounds from graphical data [5]. Similarly, some existing peripheral visualization devices also seek to represent sound intensity as a function of time in a glanceable secondary display [1]. The sound intensity graphs generated in the secondary display must also be interpreted by a user who understands the relevance of the data to his environment. A third approach to visual sound awareness is a system which displays generic, location-independent acoustic information in a digital image [6]. This generic visualization performs acoustic event recognition to identify major warnings or variations in noise rather than providing a constant stream of audio information about the user's environment. In all three of these visual approaches to sound awareness, non-spoken sound information concerning sound amplitude and pitch is consolidated into a visual graph. Our project builds on these methods of audio visualization by simplifying and incorporating the generated images into the user's natural environment to create an enhanced augmented reality experience.

2.1.3 Design Functionality

Our design will focus on capturing sound intensity, displaying captions in real time, and performing sentiment analysis. The design will also present the sound data as simplified visuals, such as colors, text, and emoticons, that are easily comprehensible and non-distracting to the

user. A basic graphical representation of sound intensity will be generated by mapping sound amplitude to the color spectrum (e.g. a loud sound flashes red while a softer sound is displayed as green). Real time captions will also be displayed using a live transcription service. Information about sound amplitude combined with live captioning will give the user a good understanding of volume and urgency as well as help the user identify who or what is making the sound. Furthermore, emotional state will be conveyed to the user in the form of emoticons. These graphical elements will be combined with the user's field of vision using a near-eye optical system housed in a head-mounted display. The combination of sound intensity information with live captions in a glanceable image that can be overlayed over the user's field of vision will provide the user with a general understanding of the sound's urgency and identity. This will enhance his ability to comprehend essential noises and dialogue and respond accordingly.

2.1.4 Near-Eye Displays

One of the main engineering challenges of this project is the modeling and execution of a see-through near-eye display. This display will project the generated images and combine them with light coming from the real world. While there are many highly engineered, different solutions for near-eye displays, most share three common elements: A micro display, optical components, and a combiner [9]. The micro display provides the light for the computer-generated image, but is generally not facing the eye. The light from the display is then manipulated with optical components – lenses, mirrors, and polarizers can be used. These components transform the light from the display so it is more palatable to the eye. Finally, a combiner is used to merge the display light with light from the environment. To fabricate an effective display for the headset, we will need to perform extensive simulation of the optical system to ensure the final image will be large, focused, and sufficiently bright. After simulation, the optical housing needs to be structurally rigid so that the optics perform consistently. The near-eye display is one of the most involved problems in this project, and its performance is crucial to the overall quality of our final product.

2.1.5 Ethical Considerations

This project raises a few ethical concerns, including reality distortion, safety, collection of user data, and intellectual property. Establishing the fidelity of a link between the physical and digital

worlds is imperative to making sure the user's reality is not distorted. Even a small lag or misinterpreted sound can have major consequences for someone who relies on the technology as an extension of his senses. Inappropriate designs can increase communication friction and elevate risk of misunderstanding social and communicative intent, leading to increased social distance [7]. Therefore, this research project will have a central focus on precision and authentication as well as utility. Additionally, the display needs to be safe for the eye. This involves making sure the amount of light from the screen that reaches the eye is consistently under the level that is deemed safe to prevent eye injury. As specified by the American National Standards Institute (ANSI) Standard for Safe Light Emission, this means that the light reaching the user's eye must be categorized as Class 1 [15]. The benchmarks for light pulse duration and wavelength are given in Appendix A. Additionally, since our project involves always-on microphones, our system needs to be transparent about what data is collected and what is not. Data collection issues are magnified when applied to wearable technologies, where the device has access to a large portion of the user's activity, schedule, and overall life. Lastly, our solution needs to not infringe on anyone's intellectual property. To mitigate this risk, we have conducted an extensive prior art search, and we will continue to monitor published work throughout the project's development.

2.2 Specifications

The wearable visualization device will perform sound localization, natural language processing, and facial tracking. It will overlay the resulting information about speaker identity, caption transcription, and sentiment data as graphical visualizations onto the user's perception of the real world. This combination of audio graphics and the user's perception of his environment will create an augmented reality experience that will enhance the user's ability to interact with others. Microphones and a camera housed in the head-mounted display will collect audio and image data from the user's environment. This information will be processed through tracking and audio processing modules to generate virtual graphics relaying audio data such as captions, color-mapped sound intensity, and sentiment emoticons. The optical system in the wearable device will receive these images and overlay the graphics on top of the user's perception of his natural environment. The entire system will be powered by a laptop computer, and it must operate in any natural combination of lighting and noise conditions.

2.2.1 Inputs and Outputs Specifications

The entire device will be powered by a laptop computer, which will also handle the processing power for the software. The use of a laptop computer was chosen over a mobile approach for mostly processing specification reasons. While mobile devices offer greater flexibility in terms of application caching and portability, laptop computers are more suited for the computational and graphics processing power that this project requires without adding the complexity of cloud computing or other distributive solutions [8]. The device will also have two microphones, a camera, and a display, which will all be powered by the laptop via USB. The final product will be evaluated with an external camera to emulate the human eye. Figure 1 below shows how the microphones, input camera, head-mounted display, laptop, and camera for demonstration will be interfaced.

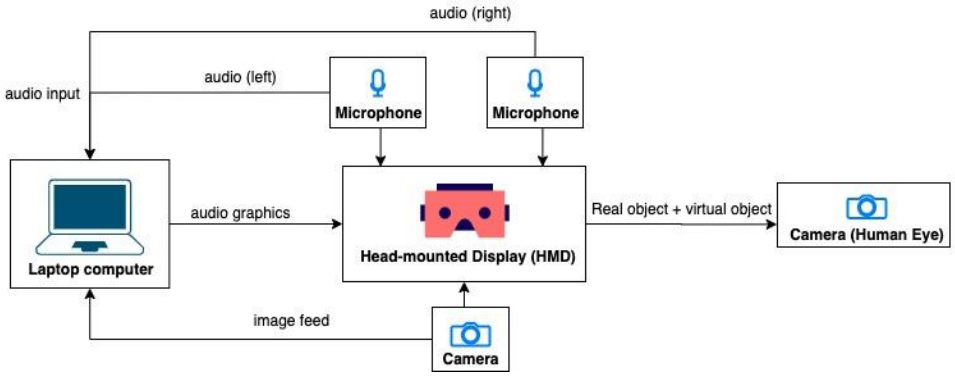


Figure 1. System configuration

2.2.2 Operating Environment

The device should be able to be used in any environment in which a person will encounter sounds. This includes environments with varying lighting and noise conditions. The audio processing software will have to take into account complicated acoustics when indoors due to the presence of strong echo. The audio processing modules will also have to account for large varieties of noise when being used in areas with very loud noises from multiple sources, such as an outdoor concert or busy street. Camera tracking will be impacted by lighting conditions. In strongly lit environments, the software will have to be trained to account for glare and high exposure. Also, the tracking will have to be trained to operate well in dimly lit situations where figures are difficult to discern. In terms of the software development environment, the team must implement the design independent of internet connection or other network dependencies in order

to ensure the technology's functionality in remote environments. The hardware components must also be properly housed so that they are protected against outdoor elements.

2.2.3 Performance Specifications

High fidelity in the performance of the design is imperative since its intended purpose is to extend the senses of deaf and hearing-impaired individuals. The device will be thoroughly tested for discrepancies in visual cues and real auditory input. Extreme sound ranges must be recognized clearly by the software to avoid missing crucial audio information. Error in camera tracking does not pose a great risk since it will only decrease the amount of free space rather than cause an obstructive placement. However, this issue can be resolved by training the algorithm against a larger and more diverse dataset of images. Precise benchmarks for software error margins will be determined in later stages of the project's development. In terms of hardware, the display must be properly interfaced and safe for the eye. All parts of the devices must be safely housed to avoid exposed components and potential safety hazards.

2.3 Deliverables

The augmented reality software will be developed on a laptop computer and interfaced with the wearable device via a DSI HSMI adapter. The device will be a head-mounted display consisting of two microphones for collecting audio data, a camera for capturing image feed, and a near-eye optical system for blending the user's perception of the physical and digital worlds. These components will be secured in a 3D printed plastic polymer housing. The final product will be composed of a laptop computer to power the system and generate graphics, microphones that will collect audio input, a camera for capturing visual input, lenses and a beam splitter to perform optical manipulation, and a monitor to display the digital images. The design will be evaluated using a camera to emulate the human eye. This system will also include a suite of software modules that will perform camera tracking, generate audio transcriptions, map sound intensity to the color spectrum, and determine emotional state from the audio data.

3.0 SYSTEM DESIGN

The entire system design will be a combination of advanced software design and optical engineering. Our software design will focus on analyzing sound intensity and sentiment and

displaying captions in real time. The design will also generate graphics to present the sound data in a simplified representation that is easily comprehensible and non-distracting to the user. These graphical elements will be combined with the user's field of vision using a near-eye optical system housed in a head-mounted display. The combination of sound intensity information with live captions in a glanceable image that can be overlayed over the user's field of vision will provide the user with crucial audio data. This will enhance his ability to comprehend essential noises and reduce barriers to communication.

3.1 Design Concept

The design will consist of software and hardware subsystems. These subsystems will interact and communicate with each other in a continuous feedback loop. The software portion will be responsible for all data processing and image generation, while the hardware portion will collect the data and output graphics to the user. Microphones and a camera housed in the head-mounted display will collect audio and image feed from the user's environment, which will serve as the input to the software modules. These modules will be stratified into a tracking module that will process the camera feed and an audio processing module that will receive the audio input. A third software module will receive the processed audio information and generate captions, color-mapped sound intensity graphics, and emoticons displaying sentiment information. Using the coordinates of free space generated by the tracking module, these virtual graphics will be rendered onto the HMD. The optical system contained in the HMD will be responsible for blending the real and virtual objects into a see-through display. A high-level breakdown of this functionality is demonstrated in Figure 2 below.

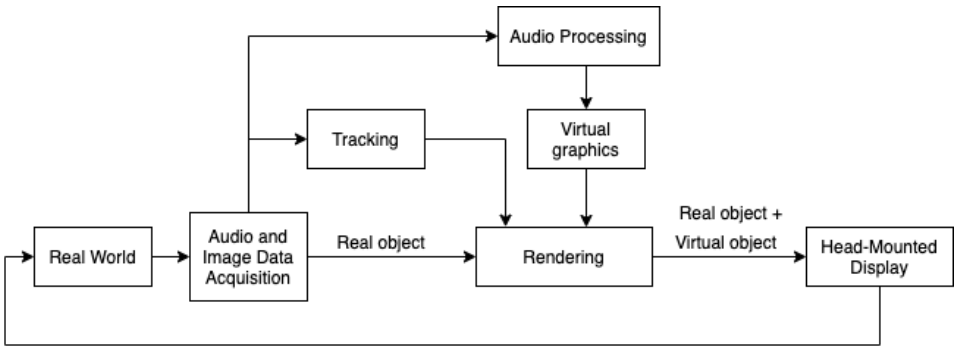


Figure 2. System Block Diagram

3.1.1 Software Modules

The software design will consist of four major classes: the device, the object detection and tracking, the audio processing, and the graphic generator. The software sequence diagram, shown in Figure 3, depicts the interactions between the user and each class. Essentially, the user will open the application, and the device class will access the external camera and microphones, getting the video and audio stream. The object detection class will detect any faces in the video frames, while the audio processor class will perform some audio separation and localization. After localization and separation, audio transcription and processing will occur. Face coordinates and audio information will be sent to the graphics generator. The graphics generator will perform sentiment analysis on the provided data and create the necessary graphics. Finally, the graphics will be displayed back to the user.

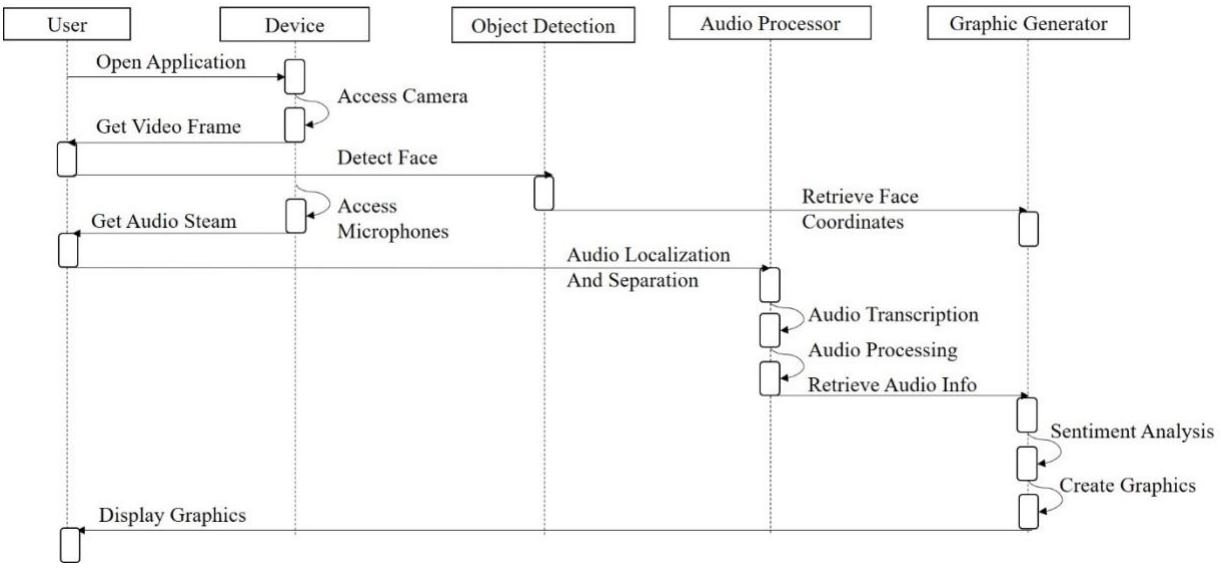


Figure 3. Software Sequence Diagram

3.1.1.1 Camera Tracking

By identifying the coordinates of faces in the user's frame of vision, we can make sure to dynamically place all of our graphics in free space. The tracking module will monitor the live camera feed recorded by the HMD and recognize key faces in the frame. The resulting coordinates pointing to free space will serve as anchor points for the audio visualization graphics. The first consideration in developing the tracking module was to decide between a

marker-based or marker-less implementation. Marker tracking trains the tracking function using specific fiduciary markers, while marker-less tracking is based on recognizing natural features. After consulting Dr. Atlas Wang, a UT professor who specializes in computer graphics and computer vision, we decided to pursue marker-less tracking. However, we must keep in mind that current marker-less tracking technologies still require a trade-off between precision and efficiency [17]. The tracking module was built on an OpenCV algorithm for face detection using Haar Cascades for its wide-ranging applications in image processing and extensive documentation [18]. This algorithm is a machine learning based approach where several interdependent functions are trained from a large dataset of positive and negative images to detect faces in other images. Developing the module to recognize faces in a frame involved training an algorithm against many large and diverse datasets and calibrating it to receive a live image feed from a camera.

The first step in developing the facial recognition module was training the OpenCV Haar Cascades algorithm with many large datasets of positive and negative images [1, 3]. We used two large datasets for training: the AT&T Facedatabase and the Yale Facedatabase. Both datasets are open-source and ethically compiled. The AT&T database contains ten different images of 40 distinct subjects with varying lighting conditions and facial expressions. It is a fairly basic dataset that provided a good reference for initial tests. Next, we improved the algorithm by training it against the Yale database, which consists of 15 people, each with 11 grayscale images sized 320 X 243 pixels. As with the first database, lighting conditions and facial expressions are greatly varied, and some faces included glasses as well. This dataset greatly improved the recognition algorithm because recognizing the faces was much more difficult.

After the function was sufficiently trained, it required calibration with the camera's specifications. Calibration involved capturing several images in various angles and mapping measured image parameters to the values that the OpenCV algorithm expected [3]. This process is not specific to the camera in use, but it does have to be done before every operation. Repeated calibration is time-consuming and prone to human error, and it could be even more cumbersome in the later stages of the project when we have to choose a camera for the design. To remedy this

calibration issue, we wrote some simple functions to take a camera calibration file input as a parameter to the module and automate randomized frame captures to streamline future testing.

When tested with a laptop camera feed, the camera tracking software performed with high fidelity. The program was able to detect all faces in the frame in various lighting and environmental conditions. This module will interact with the audio processing module to allow us to position the audio visualization graphics in free space.

3.1.1.2 Audio Processing

The audio processing class will be coded in python and have three main functions: localization/separation, intensity/amplitude extraction, and live transcription. These functions will process an audio stream in real-time and output audio data that will be received and utilized by the graphics generation class. For localization/separation, we potentially hope to use the Cone of Silence method created by the University of Washington. Their algorithm involves isolating sources within an angular region,

$$\textit{angular region} = \theta \pm w/2, \qquad (1)$$

where θ is the angle of interest and w is the window of interest [20]. By exponentially decreasing w and performing a binary search, the different audio sources can be localized and separated [20]. Once the audio sources have been isolated, the audio streams will be handled using the PyAudio library. The PyAudio library will enable us to extract features such as sound intensity and amplitude. This library will also enable us to pass the audio stream through Google Cloud's Speech-to-Text API. Google Cloud's API uses powerful machine learning models which our design will implement to create accurate live transcription of speakers.

3.1.1.3 Graphics Generation

The modules of the graphics generation class will receive information about sound intensity, sentiment, and transcription from the audio processing module and actually generate virtual visualization graphics. These graphics will then be displayed on the near-eye display via HDMI interfacing. We decided to use Unity, a comprehensive game engine, to develop the graphics.

Our main consideration in selecting Unity was its easy compatibility with our tracking software [19]. Unity is available as a plug-in to our tracking library, so we can perform viewpoint tracking and virtual object interaction simultaneously. This will help us avoid future issues with tracking and graphics generation interfacing. The platform supports the creation of 2D and 3D graphics, and it offers a primary scripting API in C#. Also, Unity is a cross-platform engine, so it is guaranteed to be compatible with every major operating system. Future testing will verify the interfacing capability of Unity within our system.

3.1.2 Head-Mounted Display

The hardware subsystem is completely encapsulated by the head-mounted display (HMD). The HMD will receive graphical data from the software system and then show it to the user. This functionality requires two major components: display interfacing and optics. The software system will output the image data in HDMI protocol. This HDMI signal will be converted into MIPI DSI using a premade IC that can be purchased online. Once the data in on a DSI bus, the LCD screen will be connected, allowing the software to print any type of image directly onto the screen. The LCD is the bridge between the software system and the optical system, as it converts the digital data into light. The optical system will then manipulate the incident light from the micro display and project it onto the user's eye. This will allow the user to not only see the real world, but also any computer-generated imagery that we wish to project on top of the real world.

3.1.2.1 Micro Display and Interfacing

To interface the software system to the hardware optical system, we will convert the digital image data from the software application into light emitting from a micro display panel. Figure 4 shows how the input digital images generated by the software modules will be converted from HDMI to DSI and displayed on the LCD panel. The HDMI to DSI converter and LCD panel are both well in our budget. The display is an LCD panel with a maximum output of 800 nits with a diagonal size of 1.54 inches.

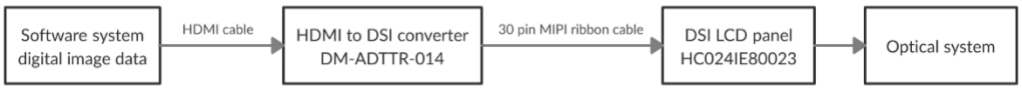


Figure 4. Software to hardware interfacing diagram

3.1.2.2 Optical System

The optical design will take in light emitted from the LCD panel and manipulate it via a series of beam-splitters and mirrors, so that it is directed onto the user's eye. The user will, at the same time, be viewing the real world through the headset. The LCD display will pass light through a polarized beam-splitter. The light will then bounce off of a curved mirror that will focus the light into an image, where it is then bounced off the same beam-splitter. The second point-of-contact the light makes with the splitter will cause the light to be reflected into the user's eye. While the user will be seeing a generated image from the display, they will simultaneously be viewing the real world, as shown in the Figure 5 below.

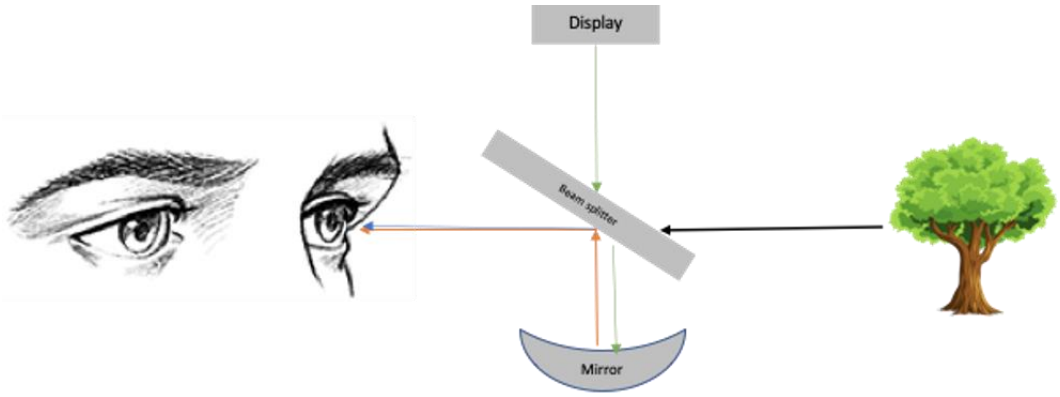


Figure 5. Optical manipulation using beam splitter and mirror

3.2 Risk Reduction

As the creation of the HMD system poses several risks, our team identified and performed 4 risk reduction experiments. These risks include potential issues in image display/interfacing, errors in the optical design, graphics obstructing user view, and inaccuracy in captioning. To mitigate these risks, the team's experiments consisted of extensive research into micro displays, simulating our optical system design in Zemax, and creating sample software modules for facial tracking and audio transcription. Through these experiments, the hardware sub-team discovered the optimal display is an LCD with 800 nits, interfaced with MIPI DSI protocol. The software sub-team concluded that OpenCV and Google Cloud's Speech-to-Text API would provide the best support and accuracy for further development.

3.2.1 Micro Display Evaluation

Before buying all of the necessary components to realize our design, our team had to be confident that the end result will meet the requirements defined in our project definition. The Enhanced Augmented Reality system needs to produce a visible image in a well-lit room. Our choice in display must support this requirement. Furthermore, the display must interface with our software system.

After researching different types of displays, LCD was chosen for this project. LCD micro displays are cheap and provide around enough light for our use case. While OLED and LCoS panels are capable of better performance, they were out of our project budget of $500. LCD panels can have a variety of interfaces in which the image data is sent through. We decided to go with the MIPI Display Serial Interface (DSI) protocol, since it can easily be converted into an HDMI signal. Finally, the team needed a goal for the brightness of the display. Brightness is critical, as it determines whether the end image will even be visible to the user. We used an initial estimation for the efficiency of our optical system (25%) to calculate the required output of the display. We found that a display of around 800 nits would be sufficient for indoor use.

According to these considerations, we determined specific parameters to narrow down the vast market of micro display panels. Any panel that meets these requirements will integrate well with both the software and optical systems of the project.

3.2.2 Zemax OpticStudio Demo

Before starting any tangible build of an optical system, the team needed reassurance that the design functioned in an optical modeling software. The angles of the light incident on the lenses had to be correct, and the behavior of light passing through the lenses had to be predictable and controllable. The main procedure in this risk reduction was building a feasible optical model that would demonstrate the behavior of ray tracing and light. At first, the team attempted to see if it would be possible to model light interacting with a beam splitter, as anticipated in the original optical planning. However, due to the team members' novel use of the software, this aspect was modeled with another surface. The surfaces in the 3D layout were able to be easily manipulated so that the ray field reflected off at the angles as planned in the original model. The model could be developed with a single ray field or multiple ray fields with each field approaching a different

focal point and tilt. Additionally, the number of rays was also up to the designer to choose from – the team chose to display the single ray field with three rays in the final presentation in an attempt to help viewers better see the travel path of incident light.

The main takeaway from this specific experiment was a better understanding of the way light behaves in an optical model. Zemax OpticStudio outputs a spot diagram that shows the resulting incident light coming out of the optical design. The diagrams show each field and wavelength combination individually, and they display how well the design has confined and controlled the spread of light passing through the system. With these results, a stable image should be able to enter the system and come out exactly as expected. The Zemax software is mainly programmed with a spreadsheet of desired lens data. Each surface is able to be manipulated to suit the desired designs. Moreover, the spreadsheet outputs useful data such as focal points and lens thickness which are quintessential for optical engineering. The modeled optical system shows light passing through a paraxial lens. This paraxial lens allows incident light to pass through and become focused to a single image about 31.4mm away. This focused image is known as a focal point in optics. At this focal point, an angled mirror was placed. This mirror bounces the light at a direct 90 angle onto a specified object. This 3D model is not indicative of our initial design, but without the results of the micro display evaluation, the inclusion of the required parts in Zemax would not be possible. Overall, this experiment was a good representation of the use of Zemax for preliminary modeling of our optical system

3.2.3 Markerless Tracking

The team discovered several existing AR captioning technologies that implemented subtitles in a fixed position or required users to configure caption placement [2, 3, 8, 10]. However, users found the captions were obstructive or cumbersome to configure [2, 3]. This would only be a minor annoyance to a customer using the product in a specific recreational setting such as a movie theater. For our design, this potential obstruction of vision has much stronger consequences because it is intended for use in a natural environment. To address this concern, our software will harness computer vision to track relevant figures in the frame and automatically reposition the text in an unobstructive manner.

This experiment involves creating a sample module that will track faces in a frame using marker-less tracking, which looks for natural features instead of specific fiduciary markers [1, 3]. This sample module will be tested with single figure, multifigure, and differently lit environments. Our intended use for the final product includes indoors and outdoors, so we tested a variety of lighting conditions in both environments. We decided to build on an OpenCV algorithm for face detection using Haar Cascades [3]. This algorithm is a machine learning based approach where several interdependent functions are trained from a large dataset of positive and negative images to detect faces in other images.

In each of the tested environments (inside, outside, single figure, multiple figures, low light, and bright light), the module was able to detect the faces and generate a bounding box around the face. Captured images of the module's functionality in the various test cases is shown in Figures 6, 7, and 8 below. However, while the algorithm was very reliable in detecting faces in the frame, it did sometimes try to identify other objects in the frame as faces that were not human figures (i.e. a button on the test subject's clothing). This misidentification does not pose a great risk since it will only decrease the amount of free space rather than cause an obstructive placement. However, this issue can be resolved by training the algorithm against a larger and more diverse dataset of images. Ultimately, the results from this experiment allowed us to select a facial tracking implementation that performs with high accuracy. This will allow us to reposition our graphics in free space and reduce the risk of obstructing the user's vision.

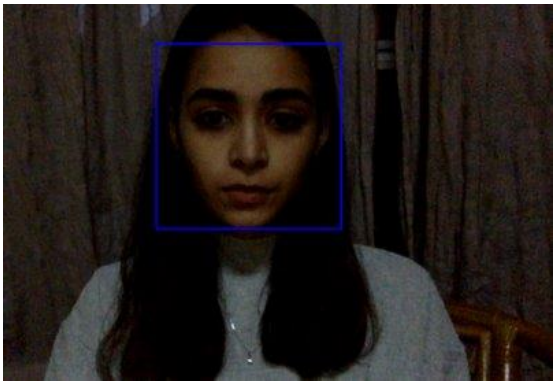


Figure 6. Face detection works well in poor lighting

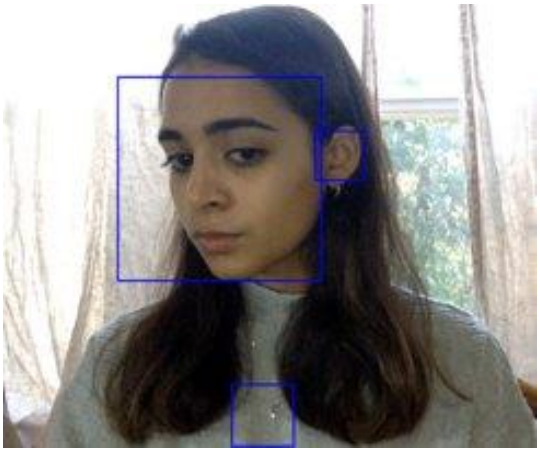


Figure 7. Some extra detection, misidentification of necklace

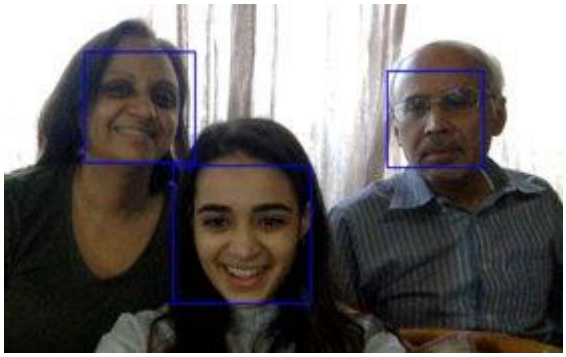


Figure 8. Multifigure detection, one figure wearing glasses

3.2.4 Real-Time Audio Transcription

While our goal includes bringing greater accessibility to hearing-impaired individuals, our technology risk hindering the user's sensory perception through inaccurate data. Specifically, inaccurate captioning could lead to miscommunication. Clearly, wrong captions imply the bad interpretation of a speaker's speech. However, as our design includes the display of emotion graphics, inaccurate captions may additionally lead to misinterpretations of a speaker's emotions. The emotion graphics will be created based on sentiment analysis. Sentiment analysis relies on natural language processing and text processing. Therefore, if we are unable to correctly transcribe speaker audio, we risk depicting the wrong emotion. To reduce this risk, the team sought to create a sample module that takes an audio stream and outputs the correct English transcription.

In conducting this experiment, the team had to research the available transcription technology and test that it produced the accurate results in both noisy and quiet environments. Through our research, we found that Google Cloud's Speech-to-Text API provides the most accurate transcriptions using powerful machine learning models. After implementing the API in a python environment, we were able to achieve inputting a continuous audio stream and receiving the corresponding text in the console.

In both the noisy and quiet environments, the sample module produced correct results in real-time. This reiterated that Google Cloud's API calls were reliable source in transcription. However, there are still further steps to accomplish in audio transcription. One issue is that the module often times-out when running for long periods of time. To fix this issue, we need to close the connection during periods of silence and reestablish the connection when someone speaks. Another potential direction is to complete some pre-processing on the audio before transcription. For example, in multi-speaker scenarios we would want to separate the audio into multiple channels and caption each speaker independently. However, with successful preliminary results there is less risk of miscommunication and the team is prepared for audio processing and creating graphics that rely on accurate transcriptions.

3.3 Test Plan

Our design will be tested iteratively, where each subsystem will be verified and then combined. From a software perspective, this means isolating each I/O block and testing the functionality of individual software modules. For example, we will only connect the microphones to the computer and test the text-to-speech and sound localization performance. Once the audio transcription module is thoroughly tested, we would move on to a different module, say the facial tracking system. Once each software module works with its respective I/O peripherals, the entire software system would then be tested. Overall, our methods of testing will allow for simultaneous development of tests, where the only blocking item will be the display interface. This will allow us to quickly debug our design and be confident that the final prototype will be functional.

3.3.1 Hardware Testing

By nature of our design, we can afford a rather clean division between hardware and software. Besides the I/O sensors, the rest of the hardware components can be tested independently of the application. First, we would verify the functionality of the display and its peripheral hardware. This means connecting the HDMI converter to a computer and the LCD panel and making sure a mirrored image is displayed. Once the screen works, we can then move to testing the optical system. A static test image can be displayed on the panel that benchmarks the visibility, field of view, and brightness of the image at the entrance pupil. Once a clear, consistent image is visible from the near-eye display, the entire design can be tested with the software.

3.3.2 Software Testing

As the software relies on input from real video frames and audio streams, most software module will be user tested in different environments to confirm functionality. Facial Tracking will be tested on robustness in different environments. Audio transcription will be tested on accuracy and latency. The graphics generation will be tested subjectively from test user's feedback and several test cases. Finally, the interactions between each module will be tested as whole.

3.3.2.1 Markerless Tracking Testing

For the facial tracking module, the testing environments will be similar to those tested in section 3.2.3 Markerless Tracking. The module will be tested in single figure, multifigure, and differently lit environments. Success of this module will be determined if faces can be quickly detected in all environments with the exception of extremely dark settings. If the module fails, the face tracking models will need to be adjusted and trained against larger datasets.

3.3.2.2 Audio Transcription Testing

The audio transcription module will be tested in noisy and quiet environments. After some calibration, the team will determine a reasonable threshold for background noise intensity that the system will tolerate. Additionally, the team will calculate the latency between the time speech occurs and the time the corresponding text is outputted. The team will minimize this latency to improve the user's understanding and reduce lag. Once the module can perform is

above the threshold for background noise intensity and latency has been reduced to only a couple of seconds, the audio transcription module can be deemed successful.

3.3.2.3 Graphics Generation Testing

The graphics generation module will be tested and judged on a both a subjective and objective basis. We will have multiple test subjects observe the corresponding graphics and assess the image quality/understandability. Additionally, we will test the obstructiveness of the graphics. The team will create several test cases in which the graphics need to dynamically move with varying coordinates/areas of obstructive space. Once we receive positive feedback from test subjects and passing test cases, the graphic generation class will be complete for integration.

4.0 PROJECT MANAGEMENT

Our team has identified the tasks that will need to take place once construction of the prototype begins and has taken initiatives to plan its completion. Each team member in their designated sub-teams, hardware and software, is responsible for at least one main task and offers support for the other tasks. The division of work is based on the team member's skills set and interests, which allow them to apply their knowledge and expand on their interests. Near the end of the project towards the final stage, all team members will synergize as a collective effort to arrive at a final prototype that is ready for presentation.

4.1 Project Activities

The Work Breakdown Structure (WBS), found in Appendix B, highlights the events that need to occur over the course of the upcoming semester. In creating the AR System for DHH individuals, the tasks will be separated into hardware and software. The main events branch from these two sub-parts. The main hardware objectives are to order parts, interface components, construct the optical system, and design the headset. The main software objectives are to complete modules for facial tracking, audio processing, sentiment analysis, and graphic creation. These main lengthier objectives consist smaller sub-tasks. For a more detailed overview of the sub-tasks, refer to the yellow highlighted section of the WBS in Appendix B. These sub-tasks should take no more than 1-3 weeks each, while the work done for the main tasks will span out over the semester.

4.2 Task Assignments

The task assignments of this project have been outlined in the Linear Responsibility Chart in Appendix C. The hardware sub-team which consists of Evan Canter, Tanjeem Mazid, and Alexander Voth will take on the tasks relating to creation of the HMD prototype. The software sub-team, consisting of Andrea Nguyen and Deepanshi Sharma, will be responsible for the creation of any software classes and modules.

For the hardware sub-team, while all 3 members will help with the construction of the optical system and display interfacing, Evan will take over interfacing the display while Tanjeem and Alexander focus on the optical system and 3D printing the housing structure. Evan has experience in embedded systems and interfacing and will serve as a good middleman when it comes to combining the hardware aspects with the software. Tanjeem has experience using CAD software to create designs from a 2D starting point and printing the designs through a 3D printer. Alec is familiar with biomedical instrumentation which will allow him to contribute successfully to the overall optics of the headset. Additionally, all members of the hardware sub-team have taken an electromagnetic engineering class which puts them at an advantage when it comes to understanding and designing optical systems and light behavior. As a joint effort, they will come together to design the final headset and make any adjustments needed.

For the software sub-team, Deepanshi will be focused on camera tracking and sentiment analysis, while Andrea's focus will be on the audio transcription and graphics generation. Deepanshi and Andrea both have a strong software background and interest, and they both have experience in software design and implementation. Deepanshi is familiar with computer graphics, as she works on a student research project involving computer vision and modeling. Andrea has taken a data science principles class which will also prove to be invaluable to the overall software sub-team goals. The division of work between the team members was coordinated in a way where their strongest skills and interests would be utilized. Together, their efforts will result in a successful project.

4.3 Project Schedule

A project schedule has been established for the timely completion of the final prototype. The team will dedicate some time over the winter break to order parts for the prototype so that by the time the semester begins, they will have the parts readily available to begin planning for the construction of the final prototype. They will begin preliminary planning for the headset so that the direction of work is clear. Once the semester begins, they will begin full-fledged work into perfecting the software aspects and constructing the actual headset. The team will check in with the faculty mentor and TAs to update them on the team's progress, before finally arriving at a presentable prototype at the end of the semester. As the end of the semester approaches, they will begin to wrap up and attempt to fully finish around a week before presenting their final prototype. For a more detailed look into to the established timeline and deadlines, refer to the Gantt Chart in Appendix D.

4.4 Project Budget

The total costs for all of the parts combined will be around $319.19, along with any shipping and handling (S/H) fees that are undetermined at this time. The team decided to order 2 extra displays and 4 extra microphones in the case that any of them become faulty or damaged in the process of building the headset. On column 4 of the Bill of Materials on Appendix E, the team highlights the main characteristics of the parts that led to their selections. Moreover, most of the parts are being ordered online from online distributors that sell optomechanical components and other engineering hardware. Column 3 of the table lists the actual component, and any other prominent identifying features for them. In total, the team managed to stay within budget and still has room for potential additions should the need arise in the course of next semester.

5.0 CONCLUSION

The design team has successfully researched, discussed, and identified the materials necessary to continue development on the augmented reality headset for the deaf and hard-of-hearing. The final design will consist of a head-mounted device housing an optical system. The system will be supported by audio processing and image processing software modules that will generate the visualization graphics. The software sub-team has acquired the means for facial recognition,

speech-to-text, and sentiment analysis. In terms of hardware, a cost-effective LCD micro display has been chosen that will interface with software via HDMI connection. Additionally, the team selected fairly cheap but quality microphones for audio collection and a 5MP camera that can produce more than sufficient image quality for tracking. Furthermore, the team performed experiments to mitigate several risks, such as incompatible micro display selection, obstructing the user's view, and potential inaccuracies in audio data. From the functionality demonstrated by the experiments, we are on the right track to develop a product that is capable of improving the lives of the hearing impaired. Adherence to the project management and testing plan, which stratify tasks between the hardware and software sub-teams, will improve our prospects for success.

REFERENCES

[1] Tara Matthews, Janette Fong, F. Wai-Ling Ho-Ching, and Jennifer Mankoff. 2006. Evaluating non-speech sound visualizations for the deaf. Behaviour & Information Technology 25, 4 (Jul 2006), 333–351. https://doi.org/10.1080/01449290600636488

[2] Dhruv Jain, Bonnie Chinh, Leah Findlater, Raja Kushalnagar, and Jon Froehlich. 2018. Exploring Augmented Reality Approaches to Real-Time Captioning: A Preliminary Autoethnographic Study. In Proceedings of the 2018 ACM Conference Companion Publication on Designing Interactive Systems (DIS'18 Companion), 4–7.

[3] Sony. Sony Access Glasses. http://goo.gl/0DKFoQ

[4] Hong H., Dewen C., Yongtian W., Sheng L. "Near-eye displays: state-of-the-art and emerging technologies." in Proceedings of SPIE -The International Society for Optical Engineering. 10.1117/12.852504.

[5] "Seven Step Strategy," United States Patent and Trademark Office - An Agency of the Department of Commerce, 22-Mar-2019. [Online]. Available: https://www.uspto.gov/learning-and-resources/support-centers/patent-and-trademark-resource-centers-ptrc/resources/seven. [Accessed: 05-Nov-2020].

[6] "Performing a Basic Prior Art Search," Performing a Basic Prior Art Search | Office of Technology Licensing. [Online]. Available: https://otl.stanford.edu/performing-basic-prior-art-search-0. [Accessed: 05-Nov-2020].

[7] H. Hu, “Devices and Methods for the Visualization and Localization of Sound,” U.S. Patent 10 111 013, 23-Oct-2018.

[8] Li Z., Mo X., Shi C., Jiang S., Jiang L. (2020) The Research of Visual Characteristics for the Deaf Wearing AR Glasses. In: Ahram T. (eds) Advances in Human Factors in Wearable Technologies and Game Design. AHFE 2019. Advances in Intelligent Systems and Computing, vol 973. Springer, Cham. https://doi.org/10.1007/978-3-030-20476-1_16

[9] “Deafness and Hearing Loss,” World Health Organization. [Online]. Available: https://www.who.int/zh/news-room/fact-sheets/detail/deafnessand-hearing-loss.[Accessed: 05-Nov-2020].

[10] Parton, B.S. Glass Vision 3D: Digital Discovery for the Deaf. TechTrends 61, 141–146 (2017). https://doi.org/10.1007/s11528-016-0090-z

[11] Bohn, D., 2017. Projection Optical System for Coupling Light to a Near-Eye Display. EP2948813B1.

[12] M. R. Mirzaei, S. Ghorshi and M. Mortazavi, "Combining Augmented Reality and Speech Technologies to Help Deaf and Hard of Hearing People," *2012 14th Symposium on Virtual and Augmented Reality*, Rio Janiero, 2012, pp. 174-181, doi: 10.1109/SVR.2012.10.

[13] Grinberg, D. et al., 2015. SYSTEM WORN BY MOVING USER FOR FULLY AUGMENTING REALITY BY ANCHORING VIRTUAL OBJECTS. US9210413B2

[14] J. Krahe, “Universal Combined System: Speech Recognition, Emotion ...” [Online]. Available:https://www.researchgate.net/publication/237605884_Universal_Combined_System_Speech_Recognition_Emotion_Recognition_and_Talking_Head_for_Deaf_and_Hard_of_Hearing_People. [Accessed: 04-Dec-2020].

[15] *American National Standard for Safe Light Emission,* ANSI Z136.1, 2007

[16] “Cascade Classifier,” OpenCV. [Online]. Available: https://docs.opencv.org/3.4/db/d28/tutorial_cascade_classifier.html. [Accessed: 24-Nov-2020].

[17] Schmidt, Adam & Kasiński, Andrzej. (2007). The Performance of the Haar Cascade Classifiers Applied to the Face and Eyes Detection. 10.1007/978-3-540-75175-5_101.

[18] “Face Detection using Haar Cascades,” OpenCV. [Online]. Available: https://opencv-Python-tutroals.readthedocs.io/en/latest/py_tutorials/py_objdetect/py_face_detection/py_face_detection.html. [Accessed: 24-Nov-2020].

[19] “Unity,” MIT. [Online]. Available: https://docubase.mit.edu/tools/unity/. [Accessed: 04-Dec-2020].

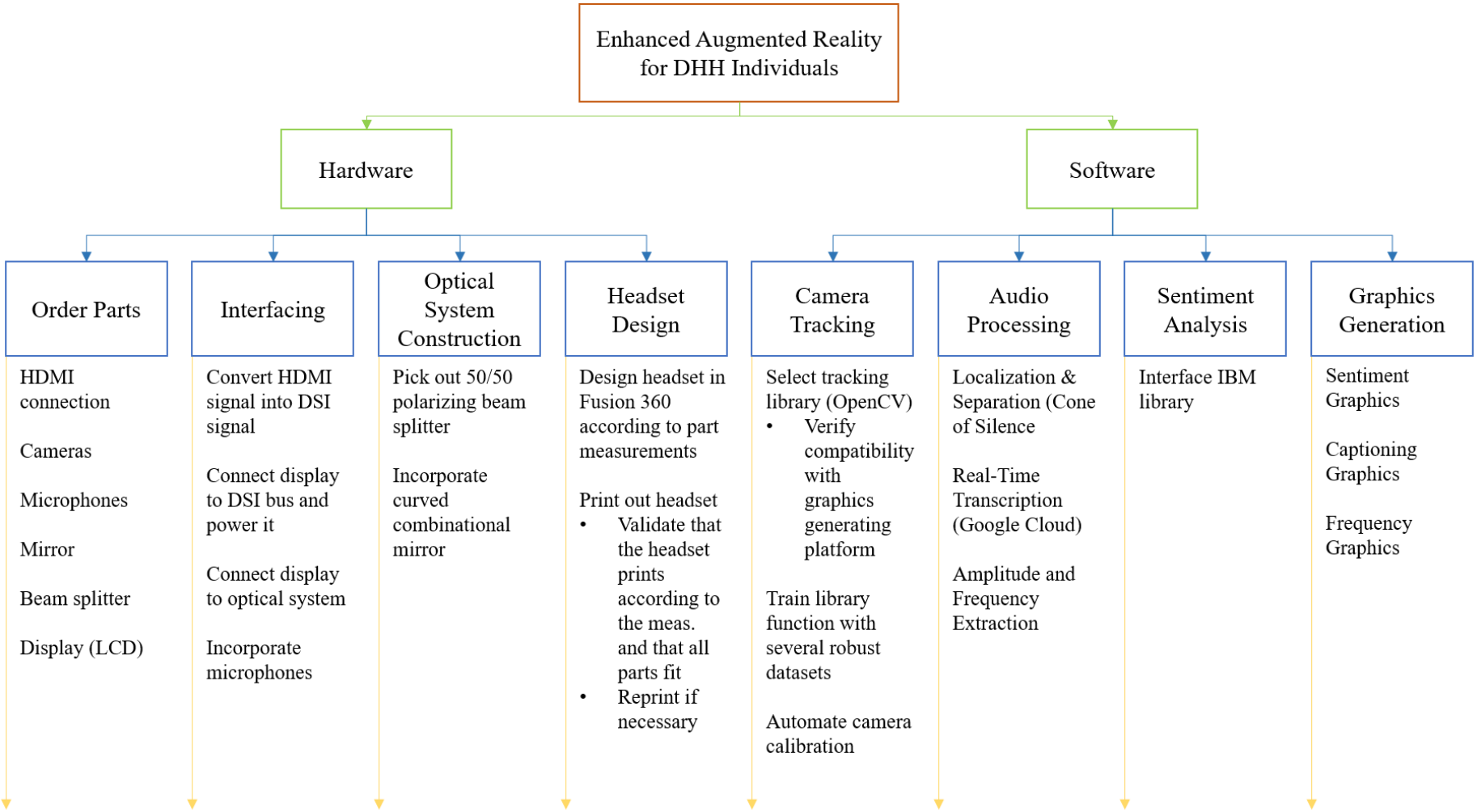
[20] T. Jenrungrot, V. Jayaram, S. Seitz, and I. Shlizerman, “The Cone of Silence: Speech by Separation by Localization,” University of Washington. [Online]. Available: https://arxiv.org/pdf/2010.06007.pdf [Accessed: 04-Dec-2020].

APPENDIX A: ANSI STANDARD FOR SAFE LIGHT EMISSION

Wavelength (nm)	Laser Type	Wavelength (nm)	Pulse Duration (s)	Class 1 (J)	Class 3B (J)	Class 4 (J)
Ultraviolet						
180 to 400	Excimer (ArF)	193	20×10^{-9}	$\leq 2.4 \times 10^{-5}$		
	Excimer (KrF)	248	20×10^{-9}	$\leq 2.4 \times 10^{-5}$		
	Neodymium: YAG Q-switched (Quadrupled)	266	20×10^{-9}	$\leq 2.4 \times 10^{-5}$	> Class 1 but ≤ 0.125	> 0.125
	Excimer (XeCl)	308	20×10^{-9}	$\leq 5.3 \times 10^{-5}$		
	Nitrogen	337	20×10^{-9}	$\leq 5.3 \times 10^{-5}$		
	Excimer (XeF)	351	20×10^{-9}	$\leq 5.3 \times 10^{-5}$		
Visible						
400 to 700	Rhodamine 6G (Dye Laser)	450-650	1×10^{-6}			
	Copper Vapor	510, 578	2.5×10^{-9}			
	Neodymium: YAG (Doubled) (Q-switched)	532	20×10^{-9}	$\leq 7.7 \times 10^{-8}$	> Class 1 but ≤ 0.03	> 0.03
	Ruby (Q-switched)	694.3	20×10^{-9}			
	Ruby (Long Pulse)	694.3	1×10^{-3}	$\leq 3.9 \times 10^{-6}$		
Near Infrared						
700 to 1400	Ti: Sapphire	700-1000	6×10^{-6}	$\leq 8.4 \times 10^{-8}$		
	Alexandrite	720-800	1×10^{-4}	$\leq 7.6 \times 10^{-7}$	> Class 1 but ≤ 0.033	> 0.033[a]
	Neodymium: YAG (Q-switched)	1064	20×10^{-9}	$\leq 7.7 \times 10^{-7}$	> Class 1 but ≤ 0.125	> 0.125
Far Infrared	Erbium: Glass	1540	10×10^{-9}	$\leq 7.9 \times 10^{-3}$		
1400 to 10^3 µm	Co: Magnesium-Fluoride	1800-2500	80×10^{-6}	$\leq 7.9 \times 10^{-4}$		
	Holmium	2100	250×10^{-6}	$\leq 7.9 \times 10^{-4}$	> Class 1 but ≤ 0.125	> 0.125
	Hydrogen Fluoride	2600-3000	0.4×10^{-6}	$\leq 1.1 \times 10^{-4}$		
	Erbium	2940	250×10^{-6}	$\leq 5.6 \times 10^{-4}$		
	Carbon Dioxide	10.6 µm	100×10^{-9}	$\leq 7.9 \times 10^{-5}$		
	Carbon Dioxide	10.6 µm	1×10^{-3}	$\leq 7.9 \times 10^{-4}$		

[a] Class 3B AEL varies from 0.033 J to 0.048 J corresponding to wavelengths that vary from 720 nm to 800 nm.

APPENDIX B: WORK BREAKDOWN STRUCTURE


Enhanced Augmented Reality for DHH Individuals
Hardware
Software
Order Parts
HDMI connection
Cameras
Microphones
Mirror
Beam splitter
Display (LCD)
Interfacing
Convert HDMI signal into DSI signal
Connect display to DSI bus and power it
Connect display to optical system
Incorporate microphones
Optical System Construction
Pick out 50/50 polarizing beam splitter
Incorporate curved combinational mirror
Headset Design
Design headset in Fusion 360 according to part measurements
Print out headset
• Validate that the headset prints according to the meas. and that all parts fit
• Reprint if necessary
Camera Tracking
Select tracking library (OpenCV)
• Verify compatibility with graphics generating platform
Train library function with several robust datasets
Automate camera calibration
Audio Processing
Localization & Separation (Cone of Silence
Real-Time Transcription (Google Cloud)
Amplitude and Frequency Extraction
Sentiment Analysis
Interface IBM library
Graphics Generation
Sentiment Graphics
Captioning Graphics
Frequency Graphics

APPENDIX C: LINEAR RESPONSIBILITY CHART

Activity \ Responsibility	Evan Canter	Tanjeem Mazid	Alexander Voth	Andrea Nguyen	Deepanshi Sharma
1. Hardware					
1.1 Display Interfacing	1	2	2		
1.2 Optical System Construction	2	1	1		
1.2 Headset Design	2	1	1		
1.3 Ordering Materials	1	2	2		
1.4 3D Printing	2	1	1		
1.5 Microphone	1	2	2		
2. Software					
1.1 Camera Tracking				2	1
1.2 Audio Transcription				1	2
1.3 Sentiment Analysis				2	1
1.4 Graphics Generation				1	2

Key

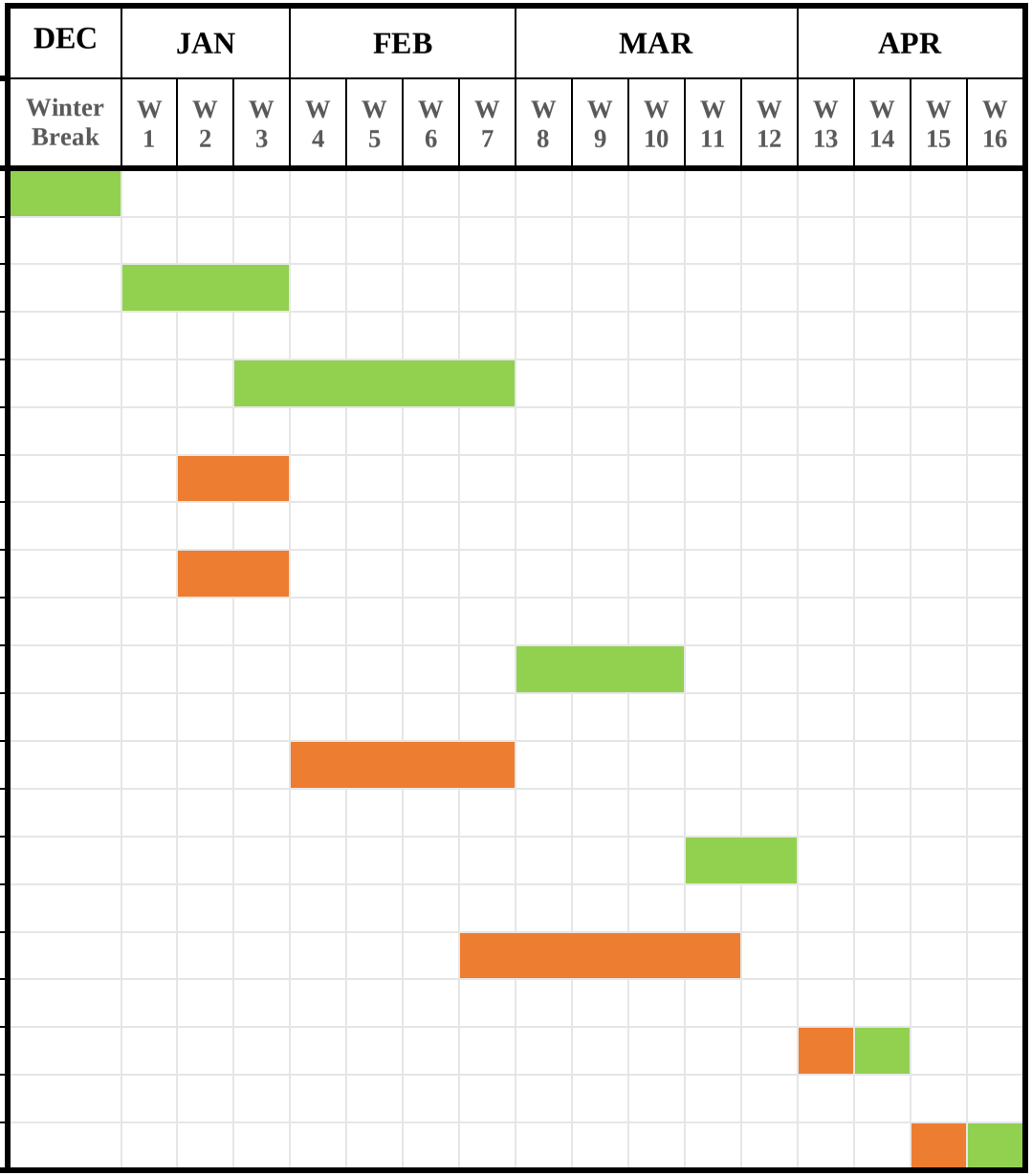
1 = Significant Support/Work

2 = Some Support/Work

APPENDIX D: GANTT CHART

Key: Hardware Software

Task	Start Date	End Date
Order Parts	12/13	1/9
Become Familiar with Parts	1/10	1/30
Interface Hardware	1/24	2/27
Improve Camera Tracking	1/17	1/23
Improve audio transcription	1/17	1/23
Design Headset	2/28	3/20
Implement Sentiment Analysis	1/31	2/27
3D Print Headset	3/21	4/3
Generate Graphics	2/21	3/27
Final Testing	3/28	4/10
Finalize Project Presentation	4/18	5/1



APPENDIX E: Bill of Materials

Qty	Type	Description	Performance Criteria	Distributor	P/N	Unit Cost	Cost
3	Display	LCD screen for HMD	800 nits 1.54 inch MIPI DSI	--	HC024IE 80023	$7.00	$21.00 + S/H
1	Optical	BSW20	1 in diameter 50/50 split beam splitter	Thor Labs	--	$106.86	106.86 + S/H
1	Hardware	Silver coated concave mirror (F = 25mm)	1 in diameter mirror	Thor Labs	CM254-025-P01	$60.34	$60.34 + S/H
5	Audio	MEMS microphone (omnidirectional) with solder pads	20 Hz –20kHz 1.62V ~ 3.6V omnidirectional	Infineon	IM69D130V01XTSA1	$1.76	$8.80 + S/H
1	Camera	Camera sensor	5 MP image capture	Digilent	410-358	$44.99	$44.99 + S/H
	Total Cost						**$319.19 + S/H**