Final Report: Voice Direction Processing System (VDPS)

ECE 492 - Capstone

Winter 2021

Evan Perret-Timms , Nicholas Serrano, Nan Ponnusamy, Luke Dela Cruz

DEPARTMENT OF
Electrical and Computer Engineering

11-203 Donadeo Innovation Centre for Engineering
9211-116 Street NW
University of Alberta
Edmonton, Alberta
Canada T6G 1H9

Revisions

Revision	Author	Changes	Date
001	Evan	Added purpose, intro and prototype discussion. Initial release.	2021-04-11
002	Nicholas Serrano	Worked on several sections of the report	2021-04-13
003	Nan Ponnusamy	Added Section 6	2021-04-14
004	Luke	Made changes to section 4	2021-04-15
005	Nan Ponnusamy	Added Section 7	2021-04-16

DEPARTMENT OF
Electrical and Computer Engineering

11-203 Donadeo Innovation Centre for Engineering
9211-116 Street NW
University of Alberta
Edmonton, Alberta
Canada T6G 1H9

Table of Contents

Acronyms

VDPS	Voice Direction Processing System
IoT	Internet of Things
PDD	Preliminary Design Document
SDD	System Design Document
ADC	Analog-to-Digital Converter
DOA	Direction of Arrival
VAD	Voice Activity Detection
VUCA	Virtual Uniform Circular Array
MUSIC	Multiple Signal Classification
UCA	Uniform Circular Array
VBPF	Voice Band Pass FIlter
GCC-PHAT	Generalized Cross Correlation with Phase Transform
UI	User Interface

11-203 Donadeo Innovation Centre for Engineering
9211-116 Street NW
University of Alberta
Edmonton, Alberta
Canada T6G 1H9

References

[1] Evan Perret-Timms, Nan Ponnusamy, Luke Patrick Dela Cruz, Nicholas Serrano, *Proposal Response: Voice Direction Processing System,* ECE 492, Dept. Electrical and Computer Engineering, Faculty of Engineering, University of Alberta, January 22nd, 2021.

[2] Nicholas Serrano, Evan Timms, Luke Dela Cruz, Nan Ponnusamy, System Design Document, ECE 492, Dept. Electrical and Computer Engineering, Faculty of Engineering, University of Alberta, March 14, 2021.

[3] Nicholas Serrano, Evan Timms, Luke Dela Cruz, Nan Ponnusam *, Preliminary Design Document,* ECE 492, Dept. Electrical and Computer Engineering, Faculty of Engineering, University of Alberta, February 4, 2021.

[4] Gcc-phat cross-correlation. (n.d.). Retrieved April 12, 2021, from http://www.xavieranguera.com/phdthesis/node92.html

[5] What is Pub/Sub Messaging? (n.d.). Retrieved April 12, 2021, from https://aws.amazon.com/pub-sub-messaging/

[6] respeaker/mic_array: DOA, VAD and KWS for ReSpeaker Microphone Array. (n.d.) Retrieved April 13, 2021, from https://github.com/respeaker/mic_array

[7] PyAudio • PyPi. (n.d.) Retrieved April 13, 2021, from https://pypi.org/project/PyAudio/

[8] S. Knudsen, Voice Direction Processing System Client Proposal, ECE 492, Dept. Electrical and Computer Engineering, Faculty of Engineering, University of Alberta, November, 2020.

[9] X. Wang, Y. Xiong and W. Huang. "An Accurate Direction Finding Scheme Using Virtual Antenna Array via Smartphones." NCBI. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5134470/ (retrieved April 14, 2021).

[10] Y. Huang. “8-Element Uniform Circular Array.” ResearchGate. https://www.researchgate.net/figure/8-Element-Uniform-Circular-Array_fig1_224130981 (retrieved April 15, 2021).

[11] Voice Band Pass Filter. (n.d.) Retrieved April 15, 2021, from https://www.circuitlib.com/index.php/schematics/product/46-voice-band-pass/category_pathway-32

DEPARTMENT OF
Electrical and Computer Engineering

11-203 Donadeo Innovation Centre for Engineering
9211-116 Street NW
University of Alberta
Edmonton, Alberta
Canada T6G 1H9

1 Purpose

This document is the final report for the Voice Direction Processing System design project. It is based on the System Design [2], which references the Preliminary Design [3] and Proposal Response [1], and the results of prototype development up to this time.

2 Introduction

The Voice Direction Processing System was a response to a proposal by Dr Steven Knudsen to create a system capable of detecting voices within a 4x4 m^2 enclosed room [3]. This system was required to be based on the Raspberry Pi microcomputer system, detect and localize sound, and provide this to the user via an API and web application. This type of system is useful for home automation companies, and can be used to enhance speaker intelligibility, in addition to assisting in other IoT related roles.

GCC-PHAT was the selected algorithm that was leveraged to localize incoming sound, and the ReSpeaker 4 Microphone Array System was used to triangulate the incoming sound sources. GCC-PHAT is based on conventional cross-correlation algorithms used to localize sound, and suited this project adequately [4]. VDPS is based on a two block system composed of a hardware unit functioning as the sound recording and processing center, and purely software backend designed to store recordings and provide simple data visualization. The frontend unit is composed of the Raspberry Pi and the Respeaker, and connects to a hosted web application.

This solution offers superior scalability and is advantageous for the end user as it provides a simple setup, and user friendly interface. One of the major goals of VDPS is to be as elegant and simple to the user as possible, while retaining as much insight as possible from the collected data. Furthermore, accuracy is of the utmost importance to VDPS, which is achieved via the GCC-PHAT Algorithm.

11-203 Donadeo Innovation Centre for Engineering
9211-116 Street NW
University of Alberta
Edmonton, Alberta
Canada T6G 1H9

3 System Design Overview

In order to meet the design goals and create a system that is easy for users to install, the system was designed into 2 main architectures. These architectures are described in greater detail in the Preliminary Design Document [3] and the System Design Document [2], but a quick mention of the functions and objectives will be stated here.

The Voice Direction Processing System Backend implemented the web application for our system. The web application allowed users to start and stop recordings happening on the hardware unit from any location, while also allowing users to perform various analyses on the recordings. Analyses included determining voice directional data, analyzing sound waves, or applying various sound filters. The Voice Direction Processing System Backend was designed such that our system could be scalable and maintained without the need to change the Voice Direction Processing System Hardware Unit. As hardware units are deployed to users, the user should only be required to perform a one time installation of the hardware. The user should not need to apply changes to their hardware unit as more features are added to the web application on the backend. By treating the software and hardware architectures as independent entities, and that changing one entity will not affect the other, this will result in a system that is relatively easy for users to install while also allowing additional features to be added to our product as time progresses.

The Voice Direction Processing System Hardware Unit focused on recording the surrounding voices within its vicinity, and maintaining a constant connection with the Voice Direction Processing System Backend to allow users to control the hardware unit from anywhere. Objectives of the hardware unit were to keep the hardware unit relatively easy to install, have it portable and easy to set up when moving to a new location, and result in a total cost of less than 100$.

There were several challenges that were encountered throughout the system design process. The first challenge was with the hardware unit and the low budget cost of each unit. Because each hardware unit had limited processing power, tasks that were performed on the hardware unit needed to be within the limits of hardware units capabilities. For example, the original Preliminary Design Document [3] stated that the processing of voice data would be done on the hardware unit concurrently as it was recording. It

11-203 Donadeo Innovation Centre for Engineering
9211-116 Street NW
University of Alberta
Edmonton, Alberta
Canada T6G 1H9

was soon however realized, that the expectation to maintain a constant connection to the live web server, while also recording data and processing the data all simultaneously was a big risk with the limitations of the hardware.

Another challenge was the actual implementation of our scalable architecture, and maintaining a constant connection between the two independent architectures. There are many variables in a forren network. For example a users network could be turned off/on at any time. Since the hardware unit and backen needed to communicate with each other, it was difficult to maintain a constant connection.

4 Prototype Implementation

In this section, we will briefly summarize each of the critical components of our prototype, and describe how the interaction between the components have changed from the Preliminary Design Document[3] as well as the System Design Document[2]. The following diagram shows the deployment diagram of our product. It has been slightly modified from the previous System Design Document[2] to incorporate the change to the processing of sound data to instead be done on the VDPS Backend rather than the VDPS Hardware Unit.

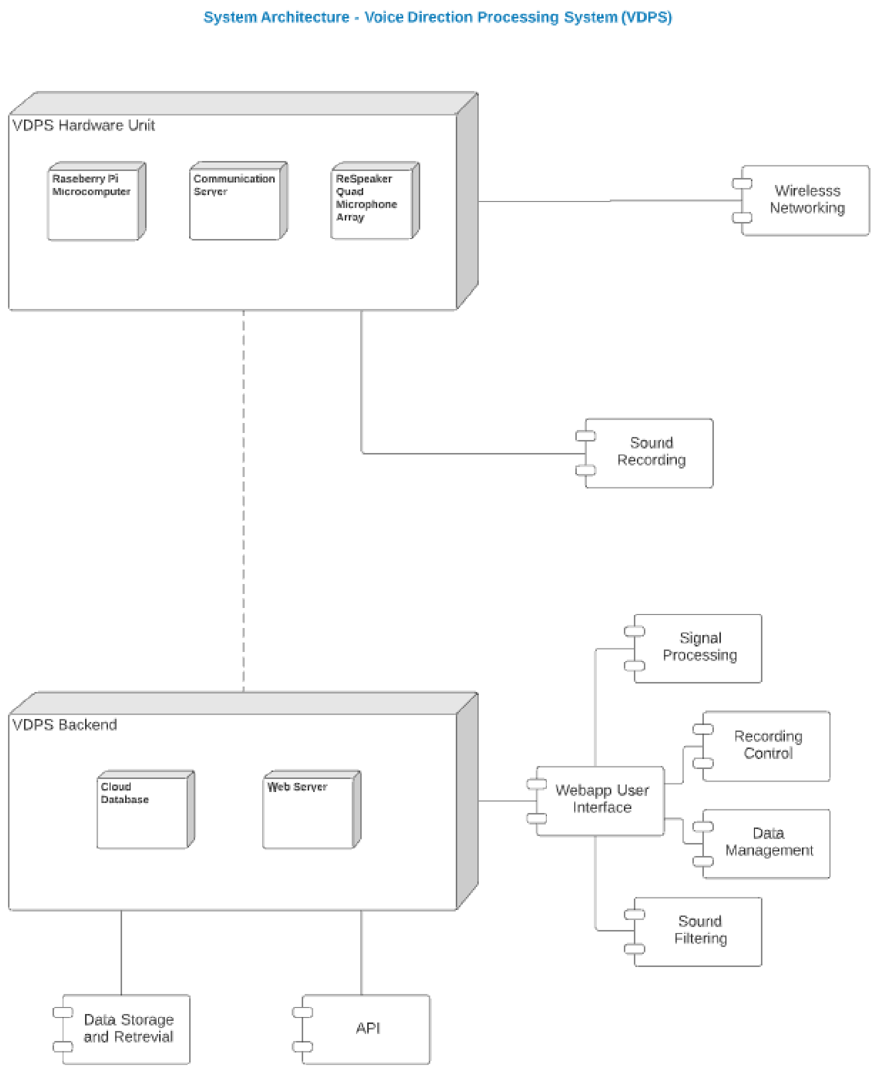


Figure 1 - Deployment Diagram Change

The next diagram below shows the Software/Firmware architecture of our system. It is largely the same as the Preliminary Design Document[3] and the System Design Document[2], with the new exception of slightly different roles for a few components.

DEPARTMENT OF
Electrical and Computer Engineering
11-203 Donadeo Innovation Centre for Engineering
9211-116 Street NW
University of Alberta
Edmonton, Alberta
Canada T6G 1H9

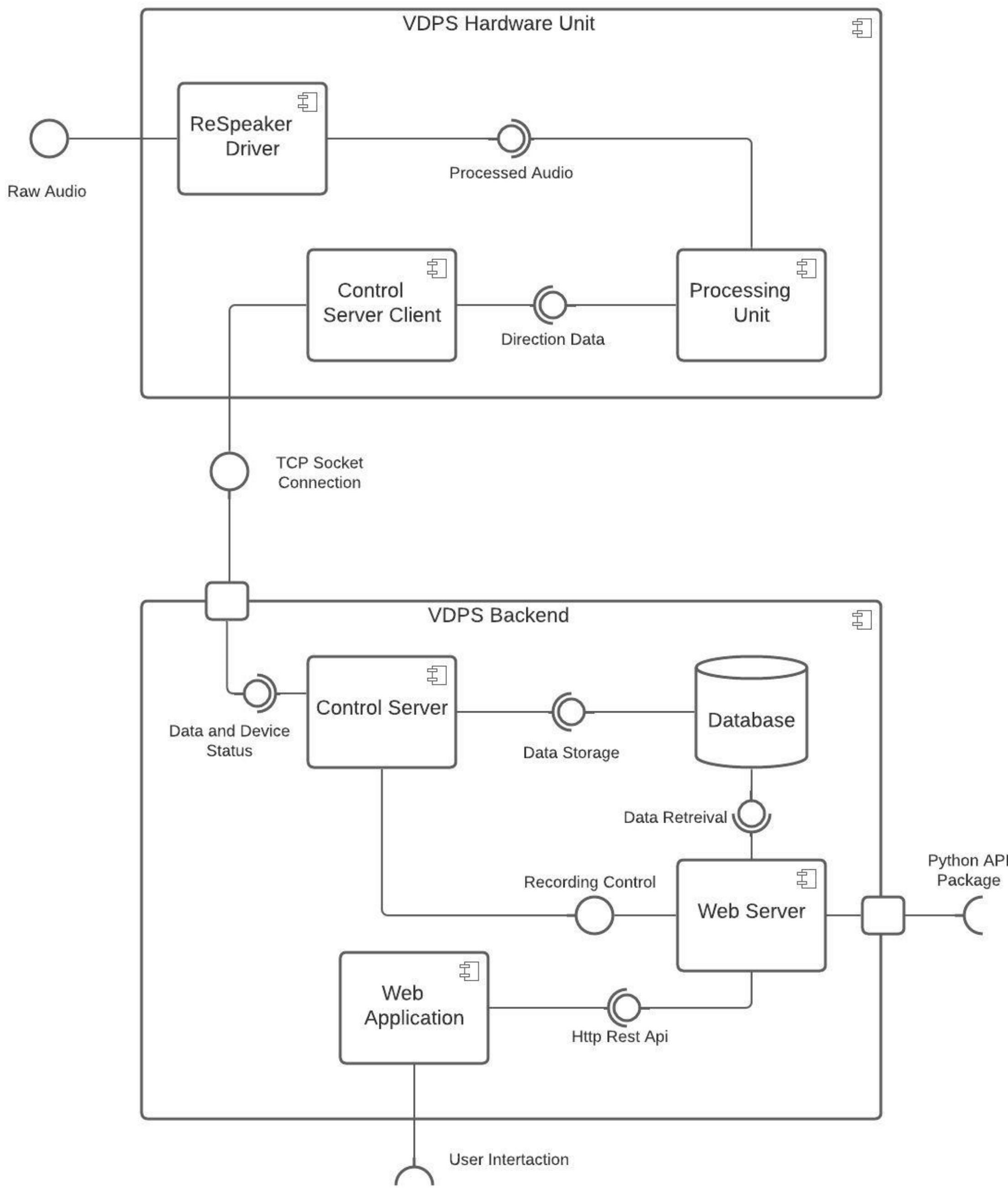


Figure 2 - Software/Firmware Component Diagram

For the VDPS Hardware Unit, much of the functionality of the components remains the same. Please refer to section 4 of the SDD [2] for a more detailed description. To summarize, the respeaker driver in combination with the installed respeaker microphone are used to record the voices of the surrounding

11-203 Donadeo Innovation Centre for Engineering
9211-116 Street NW
University of Alberta
Edmonton, Alberta
Canada T6G 1H9

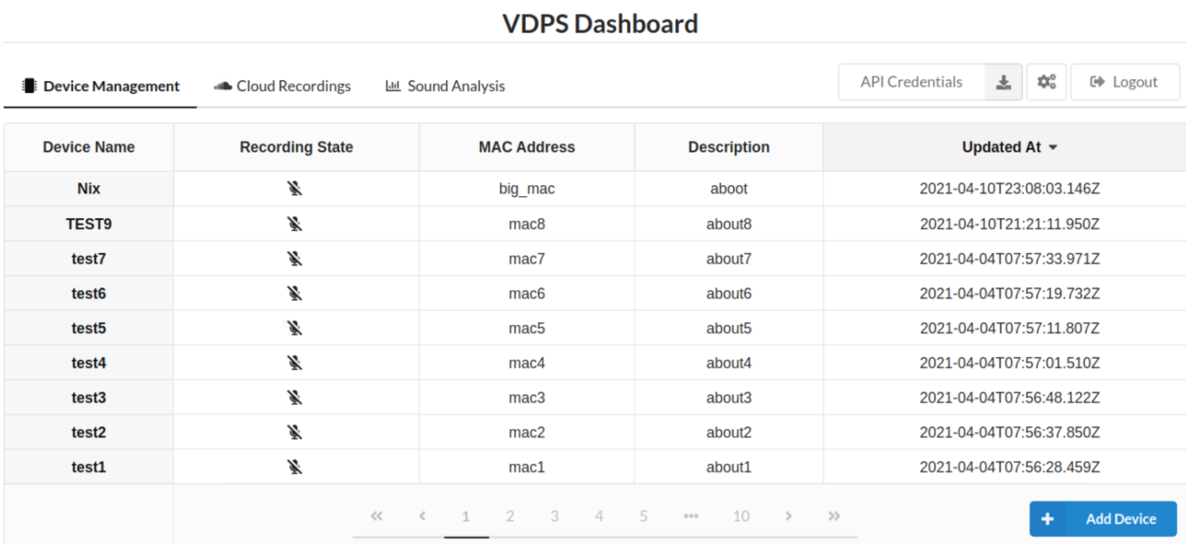
area of the hardware unit. Recordings are stored in the form of a .wav file, and are passed on to the control server client of the hardware unit. The control server client rests on the hardware unit in the form of a python script that constantly listens for commands from the VDPS Backend. A start recording command from the VDPS Backend, would cause the hardware unit to begin recording for a set amount of time. A stop command would tell the hardware unit to stop recording the surrounding area. An image of the hardware unit in its final prototype form is shown below.

Figure 3 - VDPS Hardware Unit with attached Respeaker

For the VDPS Backend, much of the functionality also remains the same as the SDD [2]. Speaking in high level terms and not talking about specifics, the combination of all the backend components in figure 2 creates a web application the user can interact with to store, view, and analyze voice recordings. The web application sits on our cloud platform, and can be accessed by any device with access to the internet. When the user logs into the web application, the user is presented with the following screen, which shows a table of all devices registered on the users account.

DEPARTMENT OF
Electrical and Computer Engineering

11-203 Donadeo Innovation Centre for Engineering
9211-116 Street NW
University of Alberta
Edmonton, Alberta
Canada T6G 1H9



Device Name	Recording State	MAC Address	Description	Updated At
Nix		big_mac	aboot	2021-04-10T23:08:03.146Z
TEST9		mac8	about8	2021-04-10T21:21:11.950Z
test7		mac7	about7	2021-04-04T07:57:33.971Z
test6		mac6	about6	2021-04-04T07:57:19.732Z
test5		mac5	about5	2021-04-04T07:57:11.807Z
test4		mac4	about4	2021-04-04T07:57:01.510Z
test3		mac3	about3	2021-04-04T07:56:48.122Z
test2		mac2	about2	2021-04-04T07:56:37.850Z
test1		mac1	about1	2021-04-04T07:56:28.459Z

Figure 4 - VDPS Dashboard

Under each device, the user can tell the device to start/stop recording, as well as view/delete old recording under that device. All data is stored in our database as indicated in figure 2. As mentioned above, each VDPS hardware unit is constantly listening for messages from the VDPS backend. The constant connection between the VDPS Hardware Unit and the VDPS Backend is what ties these components all together.

As figure 1 indicates, the processing of voice data for our prototype is now done all on the backend. To do this, the user simply navigates the VDPS Dashboard to select the device they wish to analyze. From there they select the desired recording and which sounds filters they would like to apply to the voice recording.

DEPARTMENT OF
Electrical and Computer Engineering

11-203 Donadeo Innovation Centre for Engineering
9211-116 Street NW
University of Alberta
Edmonton, Alberta
Canada T6G 1H9

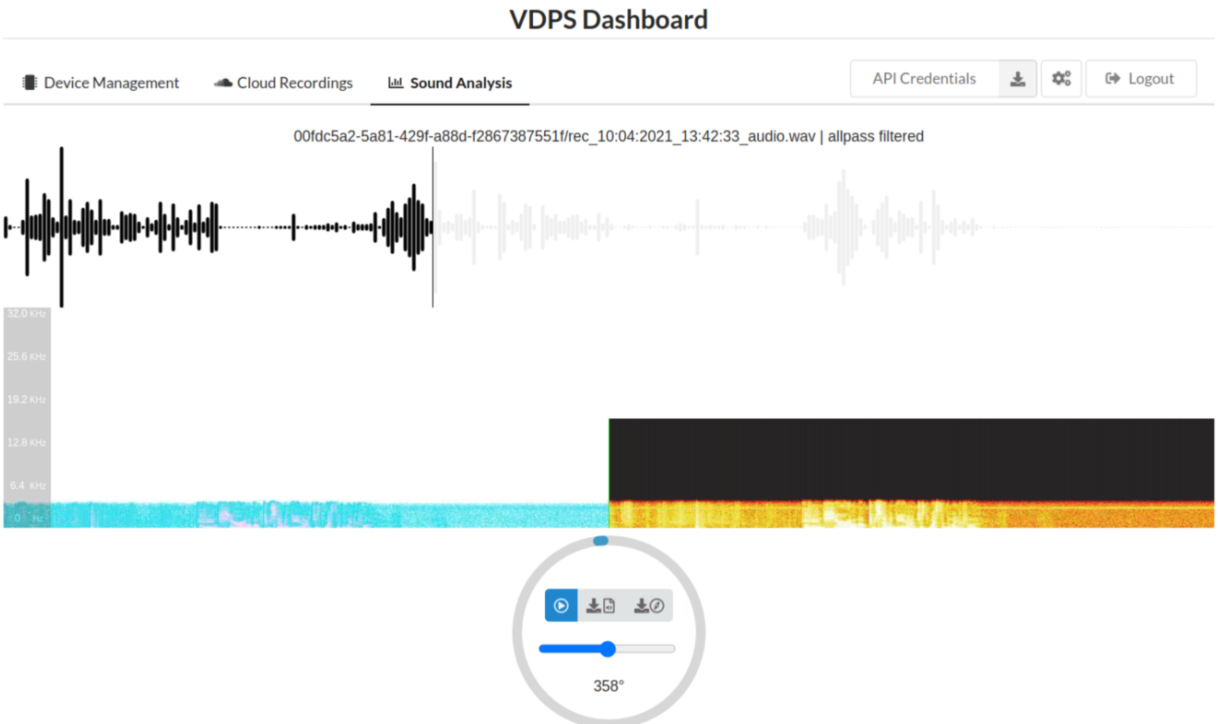


Figure 5 - Performing Voice Analysis On Web Application

Figure 5 shows the interaction between the user and the web application, in which the user is viewing the voice direction at a specific instance of time. This figure also shows the export buttons that allow the user to extract the recording/directional information as indicated in figure 2.

The final step of the prototype implementation was to test it. As required by the client, the system must be able to recognize voices within a 4x4 m^2 room. To test for this, we placed the hardware unit within a large room, and marked borders for which would resemble the perimeter of a 4x4 m^2 room. One member of the team stood on one edge of the perimeter of the room, while the hardware unit was placed on the opposite edge of the perimeter, resembling the maximum possible distance that could be between the hardware unit and a voice. The team member in the room was approximately North at 360 degrees with respect to the hardware unit.

As the prototype is required to be portable and remote controlled, another team member located at a remote location logged into the web application, and accessed the controls for that particular device in the room. The API credentials for the device were already set up by following the README

11-203 Donadeo Innovation Centre for Engineering
9211-116 Street NW
University of Alberta
Edmonton, Alberta
Canada T6G 1H9

documentation for our product, and the device was now ready for testing. After this, the team member at the remote location pressed the start recording button on the web application, and set the recording time for 25 seconds. The team member in the room began to talk, changing the tone/pitch of his voice throughout the duration of the recording.

Upon completion of the recording, the team member at the remote location then viewed the recording in the web application. Figure 5 shows the web application after performing this test. The test was found successful. The change in waveforms that were reflected were the exact times throughout the recording in which the user changed his pitch. The compass, also seen in figure 5, indicated that indeed the system was able to detect the voice at approximately 360 degrees north of the hardware unit. This test concluded that indeed the system was able record voices within the required room size, and was able to successfully determine the correct direction of the voice. This test was then repeated several times, placing the hardware unit in different locations, and having the team member in the room talking at different locations. All test results indicated that the prototype was working as intended.

5 Prototype Discussion

One of the hardest challenges associated with the creation of the VDPS prototype was the implementation of its scalable architecture. As discussed in the SDD, the VDPS hardware unit requires a remote socket server to maintain connection to the VDPS backend. For the prototype, we decided a simpler approach would be to leverage Amazon's Publisher/Subscriber services to send messages back and forth between the Pi and the backend without maintaining a constant connection. This would reduce implementation time, while offering similar functionality. The one missing piece of functionality would be the live health checks, which were skipped for the prototype.

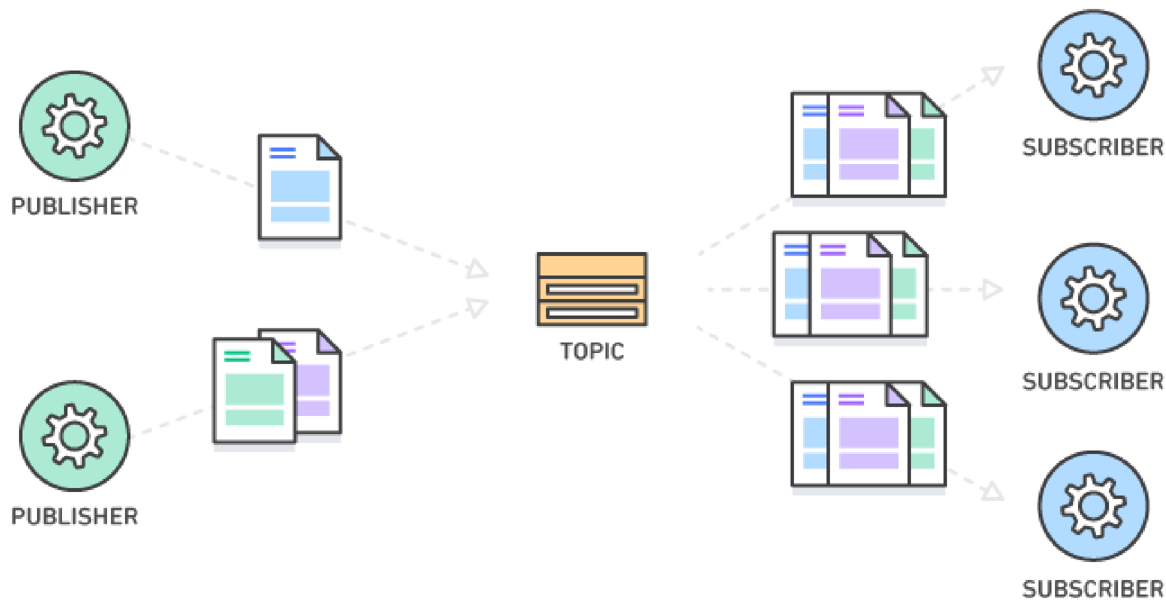


Figure 6 - Amazon Publisher / Subscriber Architecture [5]

To our surprise, GCC-PHAT [4] performed exceptionally out the box. Packaged with Respeaker Python libraries, we were able to quickly run a Python script that would allow us to print the direction of an incoming voice to the. This was sufficient for testing, but in order to integrate the processing with the remainder of our services, it was required to completely overhaul the provided test scripts and integrate them with our Amazon services. We did this by setting up Amazon's development SDK on the Pi, and authenticating ourselves via a secret and shared key (which was downloaded from the Amazon AWS console).

When it came to the development of our web application, the prototype implementation followed closely to what was defined in the SDD [2] and PDD [3]. With the exception of the backend socket server, we developed our web application and application server, in addition to the database. For the database, Amazon's Blob Storage was chosen. Blob storage is useful for large volumes of data that do not necessarily conform to what a standard SQL database would be able to provide. In this case, blob storage was useful for storing our raw audio files and direction data, which was uploaded via the Amazon SDK from the Raspberry Pis.

11-203 Donadeo Innovation Centre for Engineering
9211-116 Street NW
University of Alberta
Edmonton, Alberta
Canada T6G 1H9

Regarding development of the system, we found that it was fairly difficult to implement the system in a remote environment, specifically for the hardware unit. Nicholas Serrano proposed a solution to this, which used VNC Server to create a virtual network to allow other group members to log into the system. This is similar to SSH, but VNC Server provides an interactable stream of the device's UI, which allows multiple group members to code and work at the same time.

Finally, while the GCC-PHAT algorithm proved effective, it did require some basic filtering to improve the accuracy of the device. This was accomplished by effectively modifying the parameters in the provided code libraries from Respeaker [6]. PyAudio was the underlying interface used with the Respeaker, and a couple values were tweaked in the class construction code [7].

6 Future Work

The final product of the VDPS design project was quite successful upon review of the client's objectives for this project [8]. The client required us to create a voice direction processing system using the Raspberry Pi4 that could detect and compute the angular direction of a voice with respect to the microphone array. As stated in the proposal, we were able to successfully utilize various algorithms in our system to effectively recognize the voice recorded by our system, as well as optimize identification of vocal sources within the room. Of current, the resulting product from the VDPS project can be successfully integrated into further applications for potential use in a smart home product. Nonetheless, we'd like to explore some extensions relevant to this product that could lead to possible improvements in the effectiveness of the system and/or additional features that the client would benefit from.

Future prospects of VDPS applications in smart home products can be further explored based on the benefits achieved through integration of the product. Some ideas for applications in smart home products are listed below:

- A robotic home cleaner that is voice-activated and sensitive to direction of voice
- Home lighting system that is voice-activated based on locality of vocal activity
- Indoor and outdoor surveillance system that is responsive to vocal activity, with the capability to focus in on direction of vocal activity

11-203 Donadeo Innovation Centre for Engineering
9211-116 Street NW
University of Alberta
Edmonton, Alberta
Canada T6G 1H9

- Smart home speakers that are equipped with features such as sound boosting in response to identification of direction of vocal activity, as well as voice-activation

With that being said, depending on the application, further improvements may have to be made to the VDPS product to better suit the needs of the contextual situation. For example, application in a home lighting system would require a greater range of the voice detection capabilities of the system depending on the size of the home where it is installed. As such, we have brainstormed a few improvements that can be made to our product to account for some of these considerations. The improvements will be discussed more in depth in the subheadings to follow.

6.1 Deployment of UCA Microphones for Increased Accuracy

Exploring improvements to system capability through incorporation of multiple units in one room/space seems to hold promising advantages. By accounting for the geometry of the multiple-unit system setup, recording data with more accurate timestamps and employing cross-correlation of the input signals, it may be possible to increase the response of the system to vocal activity in the room by boosting the accuracy and speed of the voice-direction identification of the system. This concept is explored through analysis of a research topic on indoor localization scheme employed by a virtual antenna array that enables higher accuracy with respect to the direction-finding capability of the system [9].

In the research paper, the direction of high frequency sound sources are estimated by utilizing the concept behind Virtual Uniform Circular Arrays (VUCA) in an indoor setting. Essentially, the research is conducted with a smartphone that is rotated 360◦ around by an individual at a constant velocity. The setup of the smartphone-based system in conjunction with the experiment variables (such as speed of rotation and interval of sound data acquisition) imitate a VUCA to detect the direction of the acoustic signals. By incorporating various algorithms such as MUSIC (Multiple Signal Classification), they were able to accurately and efficiently pinpoint the direction of arrival of sound in various indoor environments.

11-203 Donadeo Innovation Centre for Engineering
9211-116 Street NW
University of Alberta
Edmonton, Alberta
Canada T6G 1H9

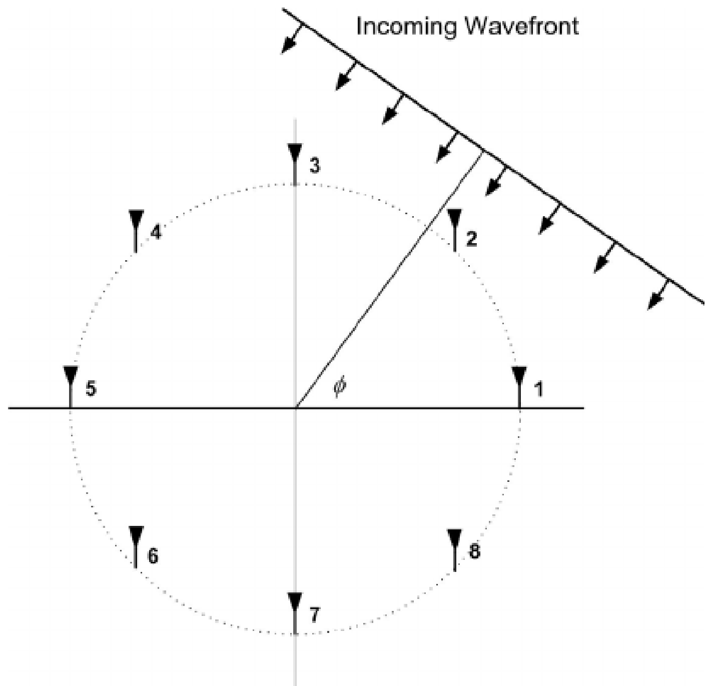


Figure 7 - 8-Element Uniform Circular Array [10]

While their findings can't be directly leveraged in our project, we can make use of the conceptual idea behind their Virtual Uniform Circular Array, and other mathematical concepts related to cross-correlation and beam-forming to enhance the accuracy of the direction finding capabilities of the VDPS system. Since our own product involves the use of the ReSpeaker, which can be thought of as a circular microphone array, the conceptual application of VUCA (depicted in figure 7) to improve voice direction-identification accuracy can be effectively leveraged. For instance, we can use a microphone array that contains an increased number of the microphones for better modeling of a uniform circular array, which exhibits numerous optimal properties for accurate sound localization based on the angles of incidence.

In addition, the MUSIC algorithm can also be leveraged to make certain improvements to our design. Due to its reliability, the MUSIC algorithm is a popular topic of research for indoor localizations. The Multiple Signal Classification (MUSIC) algorithm, which was proposed by Schmidt in 1979, enables effective estimations of the following parameters that are relevant to sound localization:

- Amount of incident wavefronts

11-203 Donadeo Innovation Centre for Engineering
9211-116 Street NW
University of Alberta
Edmonton, Alberta
Canada T6G 1H9

- Cross correlations between the incident waveforms
- Noise to signal strength ratios
- Direction of arrival

6.2 Filter Functionalities

We can also establish certain additional filtering functionalities to filter out the auditory noise that may be hindering the determination of DOA of sound data. In order to accomplish this, we can filter the signals with a VBPF (Voice Band Pass Filter) [11] to suppress any unwanted noise. Voice band pass filters are commonly used In many radio voice receivers and transmitters to keep the voice signal inside the 300-3000Hz vocal band. This could effectively filter out any noise in the sound data acquired by the microphone so that the calculation of the direction of arrival produces more accurate results.

6.3 Additional Functionality on Web Application

Another improvement we can make to our product is the inclusion of additional functionalities to the web application which serves the purpose of a client interface with our data acquisition and processing hardware unit. Example features include further improved data visualization, such as breakdown of the audio data into an intelligent fourier series for frequency analysis, and the ability to export data directly at a certain moment in time. Additionally, comparing data from two devices in parallel can be useful for multiple devices set up in the same room, which would be listening to the same source.

User friendly features include a more streamlined UX, account settings, or a dark theme to aid users while they are using the web application.

11-203 Donadeo Innovation Centre for Engineering
9211-116 Street NW
University of Alberta
Edmonton, Alberta
Canada T6G 1H9

7 Conclusions

Overall, the Voice Direction Processing System that was designed during our capstone project was successful as it effectively covered and fulfilled the client's objectives for the system. The system designed during the past 4 months of the development phase consisted of a Raspberry Pi-based hardware unit that was coupled with the VDPS backed for data collection and processing. Our final product was able to efficiently locate the direction of arrival of the voice data that was recorded by the hardware unit through processing of the vocal data by various algorithms on the hardware unit (such as DOA and GCC PHAT) as well as processing that was done by the VDPC backend.

The one goal for our product was to create an elegant and simple device that would offer insightful information regarding the recorded voice data such as the direction of arrival of the voice. Evidently our one goal was effectively captured by our final product which provided an interactive UI (VDPS Web Application) that presented valuable data regarding the various devices that were registered with the VDPS backend, as well as relevant vocal data that was recorded by each of the devices. Each recorded vocal data file could also be analyzed to acquire the associated directional data.

Potential target customers for the initial release of our product could be home owners that are looking to automate their home appliances by shifting towards the integration of an increasing number of smart home products, or companies building home automation smart devices. There is also a possibility to move towards a wider market (businesses and commercial office buildings) once the initial test-run of the product on the homeowner demographic is successful.

As for the long term, our product was designed to be scalable with the intention to eventually add new features in the future. As such, we have provided lots of documentation for our system and its codebase. The codebase is open source, allowing future development teams to download the codebase and add any additional features. Future improvements could include additional filter functionalities, improvements to the web application via adding additional features, and microphone improvements for better accuracy.

In conclusion, VDPS provides unmatched value to home automation systems through its infinite scalability, insightful features, and ease of use.