Mobile App to Allow for Multi-Source Noise Cancellation in Public Spaces

Grant Proposal

Vishal Balagani

Massachusetts Academy of Math and Science

85 Prescott St

Executive Summary

The overall aim of this project is to create a mobile app to enable portable multi-source noise cancellation for headphones. When installed, the app would connect to other phones that have the app installed, through a wifi connection. The connected phones would act as reference microphones for the headphones by transmitting the audio recorded by the microphones to the controller phone. The controller phone would then run an adapted ANC algorithm for multiple reference microphones to produce anti-noise. The resulting anti-noise would be played through headphones to achieve noise cancellation. This ANC performance should be much more effective than a regular setup with just a few microphones in the headphones themselves. It should also be able to more effectively cancel noise at higher frequencies. Both of these capabilities are possible because the headphones have a longer time to produce anti-noise.

Mobile App to Allow for Multi-Source Noise Cancellation in Public Spaces

Is it possible to make a multi-source noise cancellation setup portable? The goal of this project is to develop a mobile app that utilizes microphones on mobile phones as reference microphones for an active noise cancellation setup using headphones. Noise cancelling headphones are headphones that are meant to reduce unwanted background noise while in use. Such headphones have become increasingly popular in the past few years. With the headphone industry in the US seeing a consistent increase in volume since 2018 with a peak in 2021 the current market nearing 176 million pieces (Statista Market Insights, 2023). Many manufacturers have released products featuring ANC (Active Noise Cancellation) targeted towards the mass market in recent years and sales have increased greatly. Today it is increasingly common to see many consumers using such products daily such as Airpods Pro, Bose QC45, Sony WH-1000XM4, etc. The increase in interest in products using ANC has become apparent with the global noise-canceling headphones market projected to reach $45.4 billion by 2031 seeing an increase from just $13.1 billion in 2021 (B & K, 2022).

Active Noise Cancellation

This active noise cancellation (ANC) technology works by the concept of superposition. It involves a digital signal processing unit that can take audio, and background noise, coming from a microphone as input and produce an anti-noise signal of equal amplitude and opposite phase. When this anti-noise signal is played through the headphones the two signals combine which results in the cancelation of both noises (Kuo & Morgan, 1999). The state of ANC in headphones has improved greatly in recent years however there are still some drawbacks. ANC is incredibly efficient when attenuating lower-frequency noise but when faced with more variable higher-frequency noise it loses its efficiency (Kajikawa et al., 2012). One reason for this is that headphones have an extremely small window of time between when the input signal is received and the anti-noise signal is produced. This happens almost instantaneously every second which leaves little time to refine the produced anti-noise with the algorithm.

Lookahead

In previous research, a solution was proposed by Shen et al. in a project called MUTE. MUTE leveraged an external microphone connected to headphones through an IoT (Internet of Things) relay. This introduces the concept of lookahead since wireless signals travel much faster than sound, the headphones receive the microphone signal before the noise hits the headphones. This allows the headphones to “lookahead” into the future to know what noise it will capture before it captures it. This solves the problem of insufficient time because the algorithm can start producing the anti-noise signal before the noise reaches the headphones allowing it sufficient time to produce an accurate anti-noise signal (Shen et al., 2018). This should in theory allow the algorithm to better cancel higher frequency noise as well because it is given more time to adapt and produce the accurate anti-noise signal.

Multi-Source Noise Cancellation

In later research, in a project called WINC, this concept of lookahead was utilized and improved on. This was achieved by increasing the number of external reference microphones in the surrounding environment, as opposed to only using one, as was done in MUTE. With WINC the extra microphones allowed the headphone to sample noise coming from all directions around it and provided a much more enhanced lookahead. This resulted in a need to compensate for the extra microphones by adjusting the signal processing algorithm. The basic least mean squares (LMS) algorithm was used however it was adjusted to take all microphone sources as input. However as a result the noise canceling performance was greatly improved over MUTE and drastically improved over the industry standard (Janveja et al., 2023). The below diagram represents the configuration of the setup for WINC with multiple reference microphones signals being used as input for the DSP algorithm to produce anti-noise.

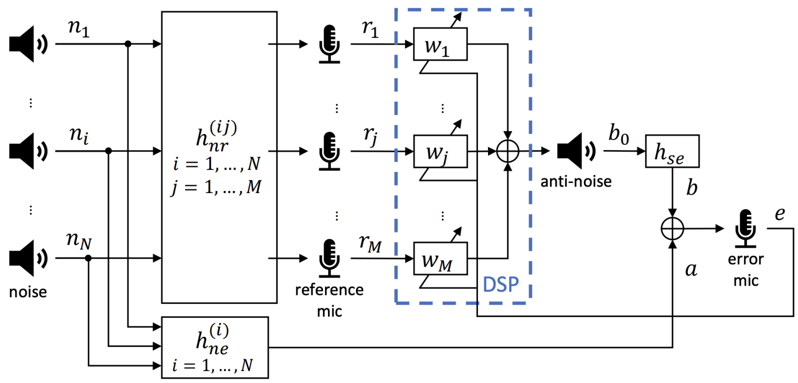


Figure 1. System diagram of WINC project showing the process of how noise cancellation works using multiple noise sources (Janveja et al., 2023).

Proposed Solution

However, as great as these solutions are, the issue with them is that they are not portable. Within the setups for MUTE and WINC external microphones needed to be set up beforehand in a predetermined environment, these microphone positions are fixed which eliminates the possibility of portability. This would make it so this enhanced multi-source noise cancellation would only work within that environment. However, the proposed solution for this project addresses that problem. If it were possible to move these external reference microphones the enhanced multi-source noise cancellation would be able to move with the user. The proposed solution uses phones within proximity of the user to act as external reference microphones. This would allow for multi-source ANC wherever the user were to go as long as there are other people carrying phones nearby. This configuration would work best in public spaces with many people such as malls or coffee shops, as there are many possible phones to connect to and make use of their microphones. As the user moves around to wherever necessary the app will adapt and connect to nearby phones to use for reference input.

Section II: Specific Aims

This proposal's objective is to provide evidence that a novel system to provide mobility to a multi-source noise cancellation system is a necessary innovation. Provided the current state of active noise cancellation in the consumer headphone market, there is currently no true active noise cancellation (Shen et al., 2018). A proposed solution utilizing lookahead and multiple microphones successfully solved this problem and produced great results with true noise cancellation (Janveja et al., 2023). However, such a solution would only work in predefined environments, lacking the ability to be adaptive and mobile. With the solution outlined in this proposal, this multi-source model could be applied to a mobile system. With the capability to work in public spaces and adapt as you move this technology could possibly be implemented in consumer technology. Such a technology would greatly improve noise cancellation in headphones and would be implemented wirelessly without much prior setup. This makes it the perfect solution for an average consumer of headphones and could be implemented in the industry by companies.

Our long-term goal is to create a user-friendly mobile app that can be installed on any given mobile phone that can wirelessly achieve mobile multi-source noise cancellation on Bluetooth headphones. The central hypothesis of this proposal is to use microphones on mobile phones to connect and relay audio data to a given pair of headphones to achieve mobile multi-source noise cancellation. The rationale is that if we increase the time gap between when the headphones take in noise input and produce an anti-noise signal a more accurate signal will be produced and played on time (Shen et al., 2018). We can increase the time gap by also increasing the number of microphones which in our case would be mobile phones to also introduce portability in our system. We can communicate audio data through compression and send it over an IoT network such as BLE or WiFi to the receiver device to sample multiple different audio sources as input (Polonelli et al.,2023). Establishing this connection and having multiple phones connected would successfully achieve the system setup needed to run the multi-source noise cancellation. The work we propose here will achieve a noise cancellation system with much better performance than the current industry standard, capable of canceling higher frequency noises as well as being portable and adaptable to wherever the user chooses to go.

Specific Aim 1: Develop a Mobile App to Record Audio on a Given Mobile Phone

Specific Aim 2: Connect the Sender Device and Receiver Device Through the Mobile App

Specific Aim 3: Allow Connection to Multiple Sender Devices and Develop Algorithm to Choose Optimal Devices

Specific Aim 4: Produce Anti-Noise Signal through ANC Algorithm

The expected outcome of this work is to have a fully functional mobile app for the end user. This user will be able to install the app and connect to his headphones via Bluetooth. This app will look for other phones with the app installed connected to the same WiFi network. The apps will then establish a connection to each other over the network. The connected phones will be used to sample and send audio data in real-time over the network to the receiving phone. The receiving phone will run the noise cancellation algorithm on the received noise signals. The processed anti-noise signal will then be played through the user's headphones. These processes will all be running in the background and the user will simply have to turn the feature on within the app.

Section III: Project Goals and Methodology

Relevance/Significance

This project is significant because it improves on a currently incomplete implementation, noise cancellation in headphones. Since the distance between the microphones and the actual eardrum on current headphones is very small, the headphones have an incredibly small time window to produce the proper anti-noise signal before the true noise reaches the eardrum, this time window is approximately 30 microseconds. If the algorithm does not meet this deadline there is a time delay in when the noise is heard by the users and when the user perceives the cancellation. This delay is the reason ANC in headphones is considered incomplete, and this effect is even worsened when dealing with higher-frequency noise (Shen et al., 2018). With the implementation of the proposed solution the distance between the user's ear drum and the reference microphones is much greater meaning the time window to produce the correct anti-noise signal is also much greater. This would effectively eliminate any time delay and would allow for better performance at higher frequencies as well making this setup a true active noise cancellation implementation. Since our implementation is also mobile it is very appealing to daily consumers and could possibly be adopted within the industry itself.

Innovation

The innovation in this project comes from the ability to make the multi-source noise cancellation system mobile. This research expands on the research previously done by Shen et al, and Janveja et al, in their respective projects. The solution they proposed brought a new system in which ANC could work at higher frequencies and be much more efficient effectively creating true noise cancellation with no delays. The proposed solution of this paper brings a novel system to apply this approach with the microphones used in mobile phones and through a WiFi connection to make the process mobile and portable.

Methodology

This project will be mainly developed in Flutter which is a code framework for developing cross-platform mobile apps. Flutter was mainly chosen due to its cross-platform functionality meaning the app should, in theory, be able to be installed on either an iOS or Android device. The app will natively be able to record audio on any phone it is installed on. This feature is necessary for any device to be used as a reference microphone and to transmit audio data to a receiving phone. The audio recorded will be chunked or compressed to be streamed over the wifi connection to the receiving phone (Amatriain & Herrera, 2001). The protocol that will be used to stream the audio data to the receiving phone will be websocket. This was chosen because it is generally considered to be a faster option for data transmission than other protocols over Internet. directly comparing it to HTTP polling requests it is almost 3 times faster (Pimentel & Nickerson, 2012). Once streams of audio are received by the receiver phone the mobile app will run the noise cancellation algorithm which is in the form of a Least Mean Squares machine learning algorithm which is a step function that updates parameters based on calculated error. Since the algorithm is taking in multiple input signals within this configuration the algorithm needs to be tweaked so it produces one anti-noise signal given multiple inputs (Janveja et al., 2023). This algorithm will be written in C++ because it is the most efficient language for digital signal processing algorithms given its low-level nature and compiler configuration (Escola et al., 2022). Once the anti-noise signal is produced it will either be sent over Bluetooth or a wired connection to the user's headphones. These headphones will also need to have a microphone within the earcup to use as an error microphone to correct the algorithm in iterations.

Specific Aim #1:

Develop a Mobile App to Record Audio on a Given Mobile Phone. The objective is to have a running mobile app with a play/pause button that records audio and saves it to the device as well as stores the chunked audio data. Our approach (methodology) is to use Flutter to create the mobile app. The Flutter “record” library will be used to implement the audio recording and chunking. The recorded audio will be saved locally on the device in its own folder. The audio will be chunked or compressed by storing our audio as a Uint8list array in Flutter. Our rationale for this approach is that this array will help us in Step 2 to transmit the audio through Websocket. This is because a Uint8list stores unsigned 8-bit integers which in this case would represent each sample of our audio data as it is being recorded.

Justification and Feasibility.

The major decision toward completing specific aim #1 was how to code or program the app needed for the purpose of the project. Multiple options were considered and compared to see which was the best for this purpose. It was decided to use Flutter for all purposes of developing the app. The main competitor that was considered was React Native. Both of these frameworks are very well developed and there are many advantages to both. Flutter was chosen because many base libraries are maintained by Google itself which allows for easier troubleshooting and, more often than not, better documentation (Wu, 2018). Flutter is also coded in the Dart programming language which is very similar to Java, a language I am much more comfortable in than Javascript, React Native's programming language. These factors are why Flutter was chosen as the framework for the app.

Expected Outcomes.

The overall outcome of this aim is to have a functional mobile app with a responsive UI for recording audio and saving it locally. It will also save the compressed audio data array. This knowledge will be used to build the base for the rest of the app as well as fuel the data for specific aim #2 in which we send the compressed audio data over wifi through Websocket.

Potential Pitfalls and Alternative Strategies.

We expect there may be some issues with the format in which the audio is compressed. This being the Uint8list, while this is a valid way to store audio data as numbers or integers the process of converting this back to an audio format may be very difficult. This process can vary based on different sampling rates and audio codecs which may vary based on several conditions. Misusing one of these could result in an invalid conversion. To overcome this, experimenting with other compression methods may be possible as well as seeing which codecs are more universally supported and seeing which sampling rates would be better for the ANC algorithm.

Specific Aim #2:

Connect the Sender Device and Receiver Device Through the Mobile App. The objective is to integrate functionality within the mobile app to send audio data between the receiver and the connecting phone. Our approach is to use Websocket as the protocol to send data over WiFi. We would set up a Websocket client on the receiver phone and each sending phone would have a Websocket client and server set up on the device. The audio data recorded by the sending devices will get sent to their own websocket server which will then establish a connection to the receiving device client. The rationale for this approach is that a WiFi connection using Websocket would be the fastest and most versatile way to wirelessly transmit the data.

Justification and Feasibility.

The approach to completing specific aim #2 was chosen after consideration of other options. WiFi was chosen as the most feasible IoT network to communicate audio data over. This was mainly because of its consistent performance, reliability, and its capability to scale. When considering other networks like Bluetooth Low Energy (BLE) many concerns arose such as transfer speed, range, throughput, and bandwidth. After deciding what network to use there were multiple protocols we could use to transfer data over WiFi. Methods such as RTSP, Websocket, and HTTP polling were all considered. Websocket was chosen because it is generally a faster and more standard option than the others. When directly compared to HTTP polling it is 3 times as fast in terms of data transfer speeds (Pimentel & Nickerson, 2012). This combination of technologies was decided to be the best option to implement our specific project with our constraints and goals in mind.

Expected Outcomes.

The expected outcome for this phase of the project is to have multiple devices having this mobile app installed. One device should be used as a host, the receiving device, and the others should be able to manually connect to the host. Over this connection, real-time audio data should be sent wirelessly to the receiving phone.

Potential Pitfalls and Alternative Strategies.

A potential downside to these methods is that these features may not have as much availability seeing as the only network used is WiFi. This is because users may not always be connected to a WiFi network, especially when they are outside. Although public WiF with many users connected is often available in places like malls, cafes, and airports, places where this technology would work the best, this may not always be the case. In this case, it would be preferable to use a connection like BLE where an internet connection is not required. However, if this were to be implemented the number of connected phones might have to be limited to address the concerns previously brought up.

Specific Aim #3:

Allow Connection to Multiple Sender Devices and Develop an Algorithm to Choose Optimal Devices. The objective is to integrate an algorithm into the mobile app which will select the phones to connect to. Our approach is to use the Received Signal Strength Indicator (RSSI) by a BLE connection to calculate the distance between the sending and receiving devices. GPS will also be used to verify this calculation and calculate the direction relative to the receiving device the connected devices are facing. The rationale for this decision is that by using both of these technologies we can have an accurate measurement of position data to feed the algorithm for optimal results.

Justification and Feasibility.

The approach to use both RSSI and GPS to determine may is necessary for this setup to get both distance and angle, essentially a vector value assuming the origin is the receiving phone. GPS is a good technology to use for larger distance calculations but when used for an application for such close proximity its accuracy diminishes. At the most precise it is only accurate up to 5 meters or 16 ft which in the context of this project is a large distance (Ali et al., 2004). This is also assuming that the GPS is always this accurate and is not lagging in any way which is common in cellphones. To accommodate for this we need to use a technology that is accurate under 16 meters, and in our case, RSSI does this perfectly. RSSI is a measure of the signal strength of a BLE connection. There is a formula to calculate distance based on this RSSI which is very accurate in an open environment in close ranges (Ali et al., 2004). However, it lacks the ability to calculate the angle relative to the receiving phone, but GPS does not and its capability to determine direction is still very accurate even at close proximity so it has been selected for this function. The combination of these technologies provides accurate measurements for all the parameters needed to feed our algorithm to select the appropriate phones for our reference microphone input.

Expected Outcomes.

The expected outcome of this phase is to have functionality within the app to have a host phone turn on the ANC feature. This will then use the algorithm developed in this phase to select the optimal phones nearby and start sampling audio and sending the data through Websocket to the host.

Potential Pitfalls and Alternative Strategies.

Potential pitfalls for this strategy may be that these calculations may be too intensive on the devices involved to be running constantly. Having these algorithms running on the sending and receiving phones constantly might be harmful to the battery life of these devices. A potential solution would be to have a trigger to only run this algorithm when the ANC performance sees a steady decrease in which case we can find new phones to connect to.

Specific Aim #4:

Produce Anti-Noise Signal through ANC Algorithm. The objective of specific aim #4 is to run the anti-noise algorithm on the received noise signals from the connected devices. Our approach is to use the algorithm referenced in the article by Janveja et al. in which they configured static multi-source noise cancelation. The rationale for this is that by using the already tested and tried algorithm there is a higher chance of better performance.

Justification and Feasibility.

The algorithm that we will be using is a modification of the industry standard model, the least mean squares (LMS) algorithm. This algorithm is commonly used in many noise-cancellation headphones and has been shown to have great performance. The algorithm we are using is a modified version of LMS which takes multiple inputs and calculates one response in return to address all the inputs (Janveja et al., 2023). This algorithm has been tested and shown to have shown much improvement over the base algorithm. Seeing as the implementations in the two projects are very similar the algorithm would also be very feasible for this project.

Expected Outcomes

The expected outcome of this phase is to have the mobile app fully functional with all of its features. First connecting to other phones then recording audio and sending in real-time over WiFi and finally producing an anti-noise signal based on this. This will effectively achieve noise cancellation and can be played over wire or Bluetooth to headphones.

Potential Pitfalls and Alternative Strategies

A potential pitfall of this setup is that to run this algorithm we need an error mic within the headphone ear cup. To effectively use a LMS algorithm you need some way of telling errors to correct your produced output. In this case, it would be an error mic that we may not have access to depending on the headphones. A possible solution to this would be to put a separate external microphone within the earcup itself and use it as the error mic.

Section IV: Resources/Equipment

To complete this project several hardware and software tools are needed. In terms of hardware, we will need multiple mobile phones to test the functionality of the app. A laptop or PC is also required to write and edit code for the project. A microphone may be needed to be used as an error microphone for the finished product. In terms of software resources the expected platforms to be used for the project are Flutter, C++, and Python.

Section V: Ethical Considerations

The concept of this project raises some ethical concerns mainly regarding digital privacy and security. Many would consider this app very intrusive as it requires access to your GPS location and the phone's microphones in the background. However, there is implied consent to this because the act of downloading this app is purely optional. Anyone downloading the app will know and have to grant permissions for this data to the app. Regardless, steps will be taken to ensure the safety of users and their privacy. All of the code used for this project will be open-source so users can see exactly how their data is being used. It will be very clear that this data will not be going to any third parties and any data stored on the device will be deleted after the action has been completed.

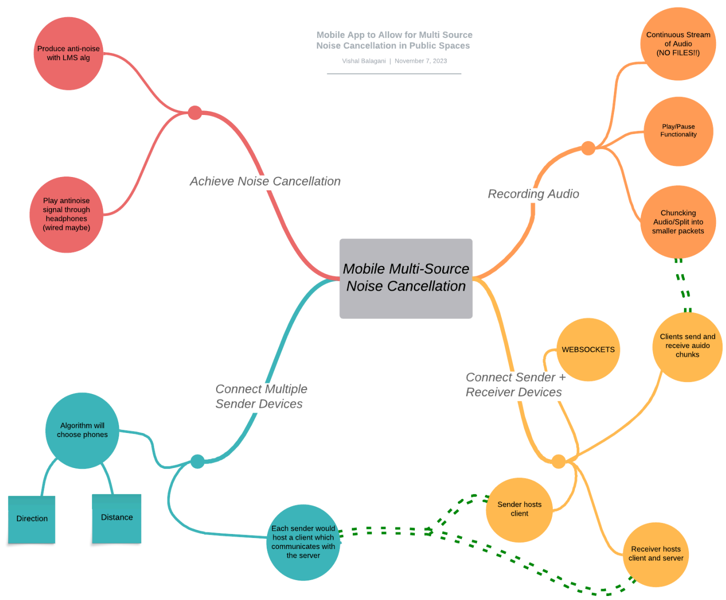
Section VI: Timeline

This project’s timeline will follow the times at which each of the specific aims is expected to be completed. Specific Aim #1 is expected to be completed by December 15th. Specific Aim #2 is expected to be completed by January 1st. Specific Aim #3 is expected to be completed by January 16th. Specific Aim #4 is expected to be completed by February 1st.

Section VII: Appendix

Mindmap

The following mindmap was made before the writing of this document. This mindmap helped plan the thoughts and ideas that would be discussed in this document. Direction and clarity of the ideas explored are apparent within this mindmap.



Section VIII: References

B, T., & K, V. (2022). Noise Cancelling Headphones Market Size and Share | 2031. Allied Market Research. https://www.alliedmarketresearch.com/noise-cancelling-headphones-market-A47198

Kuo, S. M., & Morgan, D. R. (1999). Active noise control: a tutorial review. *Proceedings of the IEEE, 87*(6), 943–973. https://doi.org/10.1109/5.763310

Kajikawa, Y., Gan, W.-S., & Kuo, S. M. (2012). Recent advances on active noise control: open issues and innovative applications. *APSIPA Transactions on Signal and Information Processing, 1,* e3. https://doi.org/10.1017/ATSIP.2012.4

Shen, S., Roy, N., Guan, J., Hassanieh, H., & Choudhury, R. R. (2018). MUTE: 2018 Conference of the ACM Special Interest Group on Data Communication, ACM SIGCOMM 2018. *SIGCOMM 2018 - Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication,* 282–296. https://doi.org/10.1145/3230543.3230550

Janveja, I., Wang, J., Guan, J., Jog, S., & Hassanieh, H. (2023). WINC: A Wireless IoT Network for Multi-Noise Source Cancellation. *The 22nd International Conference on Information Processing in Sensor Networks,* 110–122. https://doi.org/10.1145/3583120.3586964

Statista Market Insights. (2023, November). *Headphones - United States*. Statista. Retrieved November 28, 2023, from https://www.statista.com/outlook/cmo/consumer-electronics/tv-radio-multimedia/headphones/united-states#revenue

Polonelli, T., Müller, H., Kong, W., Fischer, R., Benini, L. & Magno, M. (2023)"Aerosense: A Self-Sustainable and Long-Range Bluetooth Wireless Sensor Node for Aerodynamic and Aeroacoustic Monitoring on Wind Turbines," in IEEE Sensors *Journal, vol. 23*, no. 1, pp. 715-723, 1 Jan. 2023, doi: 10.1109/JSEN.2022.3224307.

Amatriain, X., & Herrera, P. (2001, December). Audio content transmission. In Proceeding for the 2001 DAFX Conference.

V. Pimentel and B. G. Nickerson (2012), Communicating and Displaying Real-Time Data with WebSocket, in IEEE Internet Computing, vol. 16, no. 4, pp. 45-53, July-Aug. 2012, doi: 10.1109/MIC.2012.64.

Escola, J. P. L., Souza, U. B. de, & Brito, L. da C. (2022). Discrete Wavelet Transform in digital audio signal processing: A case study of programming languages performance analysis. Computers and Electrical Engineering, 104, 108439. https://doi.org/10.1016/j.compeleceng.2022.108439

W. Wu. (2018). React Native vs Flutter, cross-platform mobile application frameworks, B.Eng. thesis, Helsinki Metropolia Univ. Appl. Sci., Helsinki, Vantaa, and Espoo, Finland, Mar. 2018.

Ali, A., Latiff, L. A., & Fisal, N. (2004) GPS-free indoor location tracking in mobile ad hoc network (MANET) using RSSI, 2004 RF and Microwave Conference (IEEE Cat. No.04EX924), Selangor, Malaysia, 2004, pp. 251-255, doi: 10.1109/RFM.2004.1411119.

Shen, S., Roy, N., Guan, J., Hassanieh, H., & Choudhury, R. R. (2018). MUTE: 2018 Conference of the ACM Special Interest Group on Data Communication, ACM SIGCOMM 2018. *SIGCOMM 2018 - Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication,* 282–296. https://doi.org/10.1145/3230543.3230550

Janveja, I., Wang, J., Guan, J., Jog, S., & Hassanieh, H. (2023). WINC: A Wireless IoT Network for Multi-Noise Source Cancellation. *The 22nd International Conference on Information Processing in Sensor Networks,* 110–122. https://doi.org/10.1145/3583120.3586964

Kajikawa, Y., Gan, W.-S., & Kuo, S. M. (2012). Recent advances on active noise control: open issues and innovative applications. *APSIPA Transactions on Signal and Information Processing, 1,* e3. https://doi.org/10.1017/ATSIP.2012.4

Kuo, S. M., & Morgan, D. R. (1999). Active noise control: a tutorial review. *Proceedings of the IEEE, 87*(6), 943–973. https://doi.org/10.1109/5.763310

RTINGS. (2022, February). Noise Isolation. RTINGS.com. Retrieved December 8, 2023, from https://www.rtings.com/headphones/1-6/graph/21630/noise-isolation/sony-wh-1000xm4-wireless/16490