

## ECE4873 Project Summary

**\*\*\*Our advisor did not require project summaries as part of the course deliverables, but I saw this was required for the website, so I'm filling out to the best of my ability solely to have a file for it in our webspace.**

<b>Project Title</b>	Acoustic Source Localization
<b>Team Members</b> (names and majors)	Daniel Scarborough- Electrical Engineering
	Tiffany Ho- Electrical Engineering
	Sigong Guo- Electrical Engineering
	Ajeetpal Dhillon- Electrical Engineering
	Andrew Dulaney- Electrical Engineering
	Harry Nguyen- Computer Engineering
<b>Course &amp; Section</b> <b>Primary Advisor</b>	ECE 4873 Section A Dr. Xiaoli Ma
<b>Semester</b>	2022/Spring Course: ECE4873
<b>Web Site URL</b>	<a href="http://eceseniordesign2022spring.ece.gatech.edu/sd22p11/">http://eceseniordesign2022spring.ece.gatech.edu/sd22p11/</a>

**Project Abstract**  
(250-300 words)

RF and Wi-Fi localization has been the center of research and development for many decades, but acoustic localization remains an active area of research with algorithms like ODAS (Open Embedded Acoustic System). The objective of this project is to explore the concept of source localization for acoustic signals; evaluate, research, and devise an inexpensive, accurate algorithm that incorporates tracking and can have estimation results displayed on a flexible GUI (Graphical User Interface) This software can be applied in a classroom setting for automatic camera angle adjustment, can help strengthen various surveillance technologies, and be used in various search-and-rescue operations when visibility is limited. The project addresses the high complexity of other algorithms by utilizing a novel, less computationally expensive localization approach that utilizes TDOA (Time Difference of Arrival) and triangulation to feed angle information to be displayed on a GUI. The angle information is established through averaging multiple microphone pairs through far-field assumption and outlier filters. The core product of the project is a software algorithm that does not cost anything. To demonstrate and evaluate the project, we used a Raspberry Pi3 and its MATRIX VOICE module, which costs around 200 dollars, depending on the exact model and manufacturer and there were no other sources of expenditure. The project achieves at least 8 degrees angular accuracy in location estimation in a classroom setting and an operation range of at least 20 meters. The algorithm should also support sampling frequency no greater than industry standard 96 kHz and real time display on GUI. For future improvements, the algorithm can incorporate RF signal estimation and neural network for more sophisticated tracking algorithms.

<b>Project Title</b>	Acoustic Source Localization
List <b>codes</b> and <b>standards</b> that significantly affect your project. Briefly describe how they influenced your design.	No codes, except we tried to follow IEEE format for all written deliverables and safety guidelines
List at least two significant <b>realistic design constraints</b> that applied to your project. Briefly describe how they affected your design.	We thought about trying to get a bigger microphone array in order to pick-up acoustic signals better, but the strain this would put on the RaspberryPi would be an issue, so we just settled for improving the algorithm. We also wanted there to be a visual component to the hardware, so we added in the LED light display to complement the GUI.
Briefly explain two <b>significant trade-offs</b> considered in your design, including options considered and the solution chosen.	We really wanted to include some sort of distance feature, but we were unable to due to the complexity that would require for the algorithm, which might affect runtime, so we instead focused on making the angle measurements as accurate as possible. Also, we ended up getting rid of the signal-strength component of the algorithm, due to some problems interpreting the incoming signal. Instead, we implemented Time Difference of Arrival (TDOA) and triangulation methods.
Briefly describe the <b>computing aspects</b> of your projects, specifically identifying <b>hardware-software</b> tradeoffs, interfaces, and/or interactions.  <i>Complete if applicable; required if team includes CmpE majors.</i>	Most coding was done in Python, since it interfaces well with the hardware, which has a harder time running with C++ or other more high-level languages. MATLAB was used for some testing purposes.