Sound-to-Vibration Conversion for Deaf Accessibility

Author: Anahit Manukyan
BS in Data Science
American University of Armenia

Supervisor: Gagik Khalafyan
AUA Adjunct Lecturer

(Dated: May 9, 2024)

Abstract

The main goal of this project is to help the deaf community or people with hearing impairment to live a more productive and safe life. For solving that important problem this project focuses on two specific types of environmental sounds: road noises and emergency sirens. These are the main types of environmental sounds which are needed to warn people of potential dangers.

This project uses unsupervised machine learning models to analyze and classify those sounds. The selection of the model was made based on the experiments on available data. Since the data doesn't distinguish different types of road noise or emergency sirens, we used a clustering technique to group similar sounds without prior knowledge of their specific categories. The final model can distinguish between different types of road noise and emergency vehicle sirens by detecting patterns in the data.

The practical part of this research is embodied in a prototype glove. This glove converts recognized sounds into vibrations. The intensity and pattern of these vibrations vary by decibel. By translating auditory signals into tactile ones, the glove helps deaf people better understand important sounds in their environment. There is also a screen on the glove that shows the type and group of the given sound.

I. INTRODUCTION

In our world, it is really important and necessary to hear and respond to environmental sounds for safety and daily navigation. Individuals, who are deaf or have hearing issues often face significant challenges in listening to these signals, which most people take for granted. This limitation affects their independence, safety, and the overall quality of their lives. Recognizing and responding to sounds such as road noises and emergency sirens is not just a convenience but a necessity that alerts individuals to potential dangers.

Several traditional devices, such as hearing aids and cochlear implants, are partially effective but don't meet all needs, especially for recognizing certain types of sounds that are critical to safety. There is still a significant need for solutions that raise environmental awareness. This project will fill this gap by developing a new solution. It will go beyond the traditional auditory and use tactile response to signal important environmental sounds.

The main idea is to develop a machine learning model and create a glove prototype, which will help them to alert in their immediate environment. This approach uses the senses of touch that remain unchanged in deaf individuals, offering a new way to perceive and interpret the world around them. By converting sound waves into different vibration patterns, the glove allows users to "feel" sounds and understand different categories and groups between them.

The project employs unsupervised machine learning techniques because of the lack of specific labels regarding different types of road noises and emergency sirens. Clustering methods help in grouping similar sounds based on their features.

The goal is to enhance the interaction of hearing-impaired individuals with the environment, making it safer and more accessible. It will not only promote technological progress but also help to make modern technology accessible to all sections of society.

II. DATA

1. *Data Collection Methods*

The dataset was created to ensure a comprehensive and realistic set of sound data, which is significant for the success of the sound-to-vibration conversion project. The collection process was conducted at the National Centre for Big Data and Cloud Computing, Ziauddin University (NCBC-ZU), Karachi, Pakistan. The data was gathered using a multi-faceted approach to capture a wide range of real-world sounds. Firstly, high-definition cameras with integrated microphones were employed to record sounds in various urban settings, capturing the dynamic range of road noises and emergency sirens. Secondly, to augment the diversity of the dataset, sounds were also sourced from reliable online repositories, which provided additional samples of emergency vehicle noises and road sounds. In the end, to ensure the inclusion of clear and distinct siren sounds, emergency sirens were manually activated in controlled environments. This methodology not only enriches the dataset with high-quality recordings but also ensures a robust foundation for the development of an effective unsupervised machine learning model. [1]

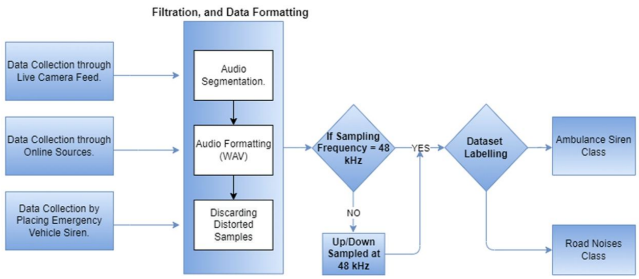


FIG. 1: A dataset was developed using two classes: emergency vehicle sirens and road noises. The complete database development process is shown below in Fig. 1.

2. *About the Data*

The final dataset contains a total of 1800 audio files. These files are into two classes, namely ambulance sound and road noise, and each of the classes contains 900 files of 3 to 15 seconds in length, as shown in Table 1.

Dataset	Developed Dataset
Emergency Siren Sounds	900
Road Noise	900
Total No. of Sample	1800
Total Duration	3.15 Hours
Audio Clip Length	3–15 secs
Sampling rate	48 kHz

FIG. 2: Table 1. Overview of the data collected

The dataset contains all the required features which can be used for classification purpose, including 20 Mel Frequency Cepstral Coefficients (MFCCs), the roll-off rate, zero-crossing rate, spectral centroid, spectral bandwidth, and Chroma_stft. They are gathered in the CSV files. These features are best known and widely used for audio recognition models.

Mel Frequency Cepstral Coefficients (MFCCs): MFCCs are numbers that summarize the way a sound changes over time. They are very helpful in identifying the type of sound. Roll-off Rate: The roll-off rate describes how quickly the sound energy decreases in the higher frequencies. Zero-Crossing Rate: This is a measure of how many times a sound wave crosses the zero point in a given time. It shows the noisiness or smoothness of a sound. Spectral Centroid: The spectral centroid indicates the center of mass of the sound's spectrum. It gives an idea of how bright or dull the sound is. Brighter sounds have higher centroid values. Spectral Bandwidth: Spectral bandwidth measures the width of the sound's spectrum around its centroid. It shows how the sound's energy spreads across frequencies, telling us about the sound's richness or sharpness. Chroma_stft: Chroma_stft stands for chroma short-time Fourier transform and represents how the energy of the sound is distributed across different pitches. It's useful for understanding the musicality of a sound.

III. LITERATURE REVIEW

a In the paper Large-scale audio dataset for emergency vehicle sirens and road noises, published 04 October 2022, presents a unique collection of audio recordings, whose aim is to assist researchers in training AI systems to differentiate emergency vehicle sounds from typical traffic noises. This dataset holds significant importance in addressing challenges related to traffic congestion, accidents. AI techniques on such data become possible to enhance traffic management strategies and improve emergency response times, particularly for critical situations. The paper also emphasizes the technical soundness and validity of the dataset, confirming its suitability for research purposes and practical applications. [1]

Nowadays, AI is in high demand in many fields and collecting a dataset is a huge task which needs a lot of work, time and resources. Vibrosight is a good way to detect actions in whole rooms using far-reaching laser vibrations. Harrison and his team talked about two ways to solve this problem at a conference in Berlin. One uses microphones, which are very common sensors, and the other uses a modern version of a listening technique used by the KGB. [2]

In the initial study, researchers tried to develop an activity recognition system called Ubicoustics. This innovative system uses microphones found in various smart devices including smart speakers, smartphones, and smartwatches. Its main objective is to identify and distinguish sounds associated with different places such as bedrooms, kitchens, workshops, entrances, and offices. Ubicoustics aims to enhance the understanding and interaction within smart environments.

Gierad Laput, a student working on his Ph.D., shared that the main idea is to use good sound collections from movies. These collections have lots of different sounds that are well-organized and perfect for training computer models. He also said that this system could be added to devices you already have with just a software update, and it would start working right away. In another paper, Yang Zhang, who is also studying for a Ph.D. in HCII, along with Laput and Harrison, talks about something which they call Vibrosight. This system can sense vibrations in certain spots in a room using laser technology. It's like the devices the KGB used in the past to pick up vibrations on shiny surfaces like windows, so they could eavesdrop on conversations causing those vibrations.

Zhang said the sensor is really good at figuring out if a device is on or off (98% accuracy) and can recognize the device itself pretty well too (92% accuracy). It can also notice movements, like when someone sits on a chair, and tell when something blocks its view, such as when a sink or eyewash station is being used.

Although researchers presented Ubicoustics, using mi-

crophones to recognize sounds in smart environments, and Vibrosight, which detects vibrations with lasernology, the following method will be the first. There has been a lot of similar research on sound recognition, but this glove-related project is the first to try to represent the specific problem and give a unique solution with Machine Learning.

IV. METHODS

This section contains the methods and comparative studies, focusing on unsupervised machine learning models used to analyze environmental sound data for assisting individuals with hearing impairments. The dataset of this study was created by merging two CSV files—one containing sounds from ambulances (Ambulance_final.csv) and the other from road noises (Road_final.csv), and the merged dataset is in (all_sounds.csv) file.

1. Data Preparation

The Data Preparation part is really important for having great machine learning models. All data from all_sounds.csv we used machine learning models, because test data can be any input sound. It was crucial to make sure the data was clean before beginning any kind of data manipulation. The dataset was checked to make sure there were no duplicates or missing values. The dataset didn't contain any missing values or duplicates. 'Filename' and 'label' columns in the dataset are not numeric, so we can't use them during the scaling process and we need to drop them. This step ensures that only relevant features are included in the model creation. Feature scaling is an important preprocessing step, especially in unsupervised machine learning. In this project, we scaled all numeric features to have a mean of zero and a standard deviation of one. After scaling, the processed data was saved into a new CSV file (scaled_features.csv) for further analysis.

2. Model Selection

In this project, the challenge was to effectively analyze and classify sounds. The complexity and variability within these sound categories, where an ambulance siren might vary significantly, and road noise contains a range of different sounds. The most suitable approach for this is unsupervised machine learning. These methods are useful at uncovering hidden patterns in data without relying on predefined labels. This is a good approach, since the precise categorization of each sound was not initially known.

Why Unsupervised Learning?

Unsupervised learning models are particularly beneficial in the cases where the data does not come with labels. This means the models have to identify patterns and structures in the data without any known outcomes. This approach is good, because it allows the models to discover the inherent groupings within the sounds, which are not immediately apparent. Clustering analysis is one of the unsupervised machine learning techniques, which I'll use in my further analysis.

The most popular and used clustering methods are K-means, Hierarchical and DBSCAN (Density-Based Spatial Clustering of Applications with Noise). So, for this project I tried to use these 3 algorithms.

K-means Clustering

K-means is a popular and prototype-based clustering algorithm known for its simplicity and efficiency. It works by separating the dataset into K distinct clusters based on feature similarity. For this project, K-means is used to group sounds that are similar to each other. This method is particularly useful for identifying patterns in the sound data, such as distinguishing between general types of road and ambulance noises. However, its effectiveness depends on choosing the right number of clusters.

Hierarchical Clustering

Hierarchical clustering is another method that builds a clustered tree, which is known as a dendrogram. This method doesn't require the specified number of clusters. Instead, it allows us to explore the data at different levels of granularity, which can show complex and layered patterns in the sounds. In this case, hierarchical clustering provides a visualization into how sounds are grouped and can show us how these groups are located within larger categories. This will offer a more detailed understanding of the soundscape.

DBSCAN (Density-Based Spatial Clustering of Applications with Noise)

DBSCAN is another clustering method, which doesn't require a specified number of clusters and works based on density. It groups together the high data density areas and the rest of the low density areas treated as a noise. So, with this method we'll have some outliers, but it is okay in this case. The possibility that It will count in the same group of the different categories of the sounds is low.

3. Model Evaluation and Hyperparameter Tuning

For development of the models and for finding the optimal one we need to do model evaluation. For K-means and Hierarchical we have n_clusters=k parameter, where k is the predefined number of clusters. In case of DBSCAN, it has two main parameters: eps, an argument name from scikit-learn, is the maximum distance between two points (for one to be considered as the neighborhood of the other), and min_samples, the number of points required to form a region. We need to define those param-

eters before doing model evaluation. During the parameters defining process it is necessary to do it with Hyperparameter Tuning, because the value of the parameters must be justified and we can't do it randomly. With the hyperparameter tuning we got the following parameters for the clustering techniques:

Best K-means configuration: 23 clusters

Best Hierarchical Clustering configuration: 20 clusters

Best DBSCAN configuration: eps=2, min_samples=2 (in case of DBSCAN, I did even the manual checking of the parameters to find the best option for not having too large or too small clusters)

After finding the best parameters for the unsupervised machine learning model techniques, it is important to fit those models and evaluate them based on several parameters. For model evaluation the following metrics were chosen: Silhouette Score, Davies-Bouldin Index (DBI), and Calinski-Harabasz Index (CHI). These metrics provide comprehensive insights into the quality of the clusters formed by the algorithms.

1. The Silhouette Score is appropriate for clustering evaluations as it measures how similar an object is to its own cluster compared to other clusters.

2. The Davies-Bouldin index signifies the average 'similarity' between clusters, where similarity is a measure that compares the distance between clusters with the size of the clusters themselves.

3. The Calinski-Harabasz Index measures the ratio of the sum of between-clusters dispersion to within-cluster dispersion.

K-means Clustering
Silhouette Score = 0.15188085514609767
Davies-Bouldin Index = 1.8203102319917974
The Calinski-Harabasz Index = 112.3538627395696

Hierarchical Clustering
Silhouette Score = 0.13244078029459785
Davies-Bouldin Index = 1.9260036160332195
The Calinski-Harabasz Index = 107.89047127270214

DBSCAN (Density-Based Spatial Clustering of Applications with Noise)
Silhouette Score = -0.22714433288340433
Davies-Bouldin Index = 0.023833637940708835
The Calinski-Harabasz Index = 100679.26091028754

The Silhouette Score tells us how similar an item is to its own cluster compared to other clusters. A high score is good. K-Means and Hierarchical Clustering have positive scores, which means they did a good job at grouping similar things together.

The Davies-Bouldin Index is about how separate and distinct the clusters are. Lower numbers are better here. DBSCAN has a very low score, much lower than K-Means and Hierarchical, which means it was better at keeping different groups apart.

The Calinski-Harabasz Index is like a rating of how well-separated the clusters are and how tight the clusters are internally. Higher numbers are better. DBSCAN has a very high score compared to the others, which would normally mean it did a fantastic job at creating distinct, tight groups.

Now, putting it all together: DBSCAN is a winner!

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) Based on the hyperparameter tuning and model evaluation DBSCAN is the most optimal clustering algorithm for this case. With the 'eps=0.2' and 'sample_min=2' parameters DBSCAN is a winner and does clustering better then other methods. With these parameters DBSCAN gives 149 clusters.

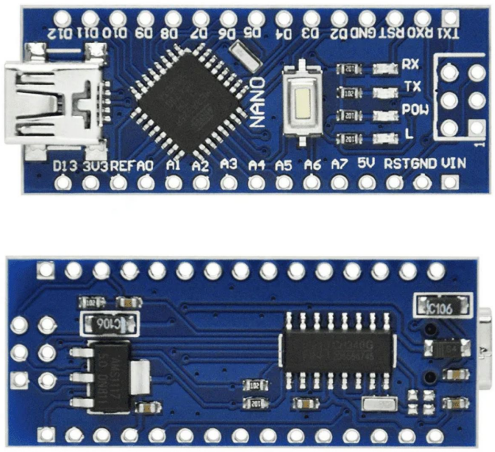
4. Mean Feature Values

After fitting the DBSCAN, there are 149 sound groups. For completing the model, it's needed to have mean feature values for each group. For example, in [Group7] we have [ambulance639.wav], [ambulance286.wav] flies and we need to have the mean feature values for that specific group. Those kinds of values are located in the mean_feature_values.csv file, which will be used for further analysis.

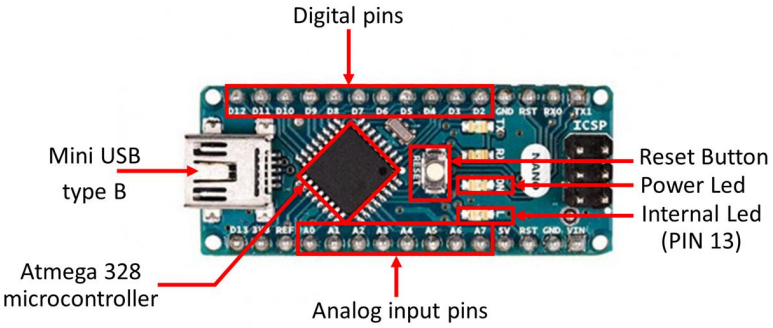
V. PROJECT COMPONENTS

In the project's engineering part these components were used:

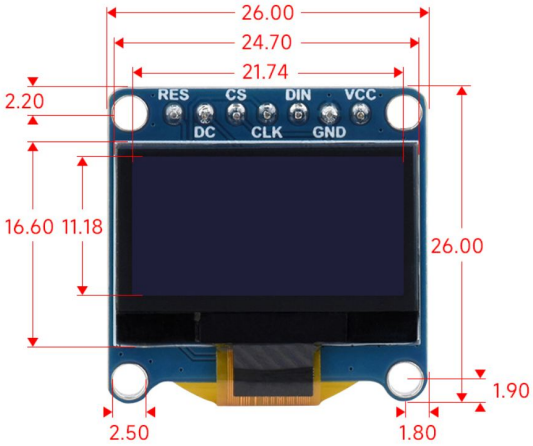
1x Arduino Nano
1x 0.96 OLED display
1x L293D Motor Driver
5x DC Motors
3D models for keeping the components and having a comprehensive structure



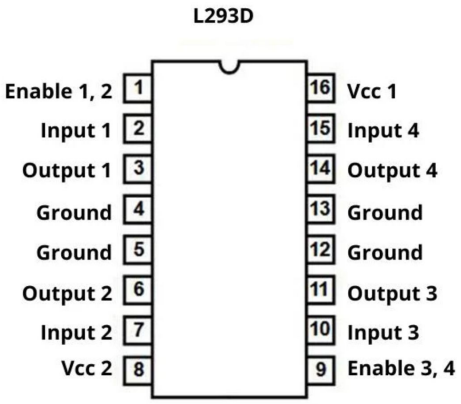
Arduino Nano



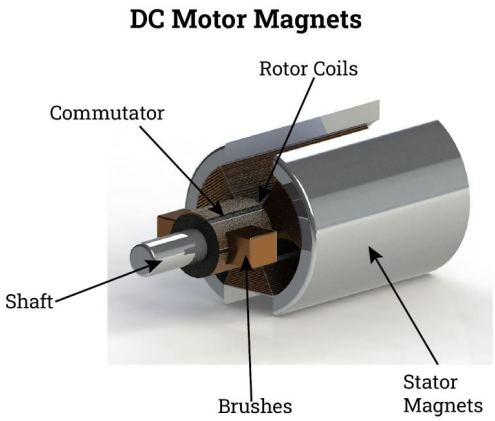
The structure of Arduino Nano



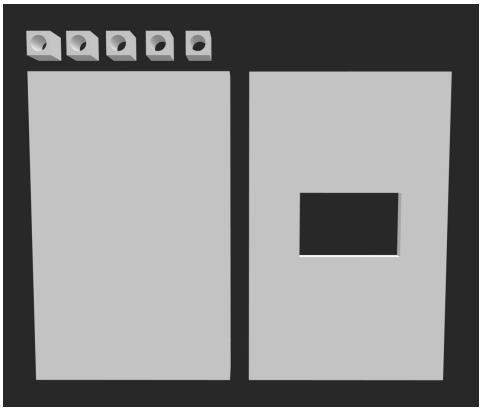
0.96 OLED display



L293D Motor Driver



DC Motor



3D models

Connections:

The Arduino Nano has multiple pins available. These pins can be analog or digital. Analog pins can send signals of 0 or 1, representing off or on states. Analog pins can also send analog signals ranging from 0 to 255, where 255 corresponds to 3.3 volts. **Connection between Arduino and OLED display:**

The OLED display is connected using I2C communication. The display has 4 pins: VCC (+), GND (-), SCL (Serial Clock), and SDA (Serial Data). Accordingly, it has been connected to the Arduino's pins: 5V, GND, A5 (SCL), and A4 (SDA). The "OLED_I2C" library is used for displaying the data.

Connection between Arduino and Motor Driver:

The motor driver is used to control the speed and direction of motors. The L293D Motor Driver allows control of two separate motors, but in this case, all 5 motors should work identically, hence all are connected to the driver. The pins on the driver are: EN1 (Speed), IN1 (First direction), OUT1 (Motor M+), GND (-), OUT2 (Motor M-), IN2 (Second direction), VCC (+)). From the Arduino Nano, pins used are: 5V, GND, 9 (Digital Pin), 10 (Digital Pin), and A1 (Analog pin, for speed).

Workflow:

1. The program starts by recording a WAV file, capturing the sound of the surrounding area.
2. From this recorded file, a db.txt file is generated containing the decibel levels of the sound.
3. The db.txt file is then used to adjust the motors' speed based on sound loudness.
4. Subsequently, the machine learning algorithm begins its operation.
5. It takes the file path as an argument, generates features based on the WAV file as a vector, and compares this vector with the data the model has been trained on. (based on the .wav to numeric values code, it generated 25 features of the input sound as we have in the training dataset)
6. Finally, it returns a group and a label if the sound is similar to something "Group_N; Type:...", otherwise "Other".

After these steps, the program initiates serial communication (UART/USART) between the computer and the board. If successful, it sends the label and speed values. The label is sent first, followed by integers representing motor speeds. As vibro motors are not available, they are simulated by changing the direction of each motor after receiving each speed value.

VI. RESULTS

In this project a glove was created, which converts sounds into vibrations and even provides the type and the group of the sound. With this glove the deaf community or people with hearing impairment can easily use this product and make their life more comfortable and safe. This project solves these issues and helps them feel confident and a complete part of the environment. With the clustering machine learning model the glove can understand the sound of the environment and helps the target community to feel the sound.

VII. FUTURE PLAN

For the future plan, the project will gather more sound data from different categories, not only from ambulance sirens and road noises. It is really important to make their daily life easier and more comfortable. The second stage of the future plan will be the model and product development. I'll try to change the size and structure of the product to make it more optimal and easy to use. Probably, I'll change the glove and try to make a bracelet, which is lighter and more comfortable for everyday use.

VIII. ACKNOWLEDGMENTS

I would like to extend my gratitude to the American University of Armenia. Specifically, thank the director of the Prototyping Laboratory Professor Zeytunyan for allowing me to use the 3D printer for my 3D models. Also, thank Vrezh Mikayelyan for the support and for helping me to work with Arduino.

[1] Asif, Muhammad, et al. "Large-Scale Audio Dataset for Emergency Vehicle Sirens and Road Noises" Scientific Data, vol. 9, no. 1, 4 Oct. 2022. https://www.nature.com/articles/s41597-022-01727-2#Sec6.

[2] University, Carnegie Mellon. "Sound, Vibration Recognition Boost Context-Aware Computing - News - Carnegie Mellon University." Www.cmu.edu, 17 Oct. 2018, https://www.cmu.edu/news/stories/archives/2018/october/sound-vibration-recognition.html

[3] Raschka, S., & Liu, Y., & Mirjalili. V. *Machine Learning with PyTorch and Scikit-Learn.*